LREC 2026

Workshop on Structured Linguistic Data and Evaluation (SLiDE)

Workshop Proceedings

Editors
Erhard Hinrichs, Joakim Nivre, Petya Osenova
James Pustejovsky, Claus Zinn

May 11, 2026

Proceedings of the first Workshop on Structured Linguistic Data and Evaluation (SLiDE)

These proceedings are available from the ACL Anthology at https://aclanthology.org.

ISBN 978-2-493814-73-9

Message from the Workshop Organizers

In the last ten years, significant advances in deep learning models and the development of Large Language Models (LLMs) have revolutionized the fields of computational linguistics (CL) and natural language processing (NLP). In turn, this has led to a complete re-assessment of the language resources and evaluation practices necessary for training LLMs and analyzing their outputs. In particular, the availability of very large amounts of unstructured data for training foundational models has come into focus, while the value of high-quality structured linguistic data with rich annotations at various levels of linguistic analysis has been downplayed by comparison. However, as CL and NLP practitioners engage further with LLMs and debate their strengths and weaknesses, the importance of high-quality, structured linguistic data has been re-emphasized.

The workshop can be seen as related to the Treebanks and Linguistic Theories (TLT) conference series and the more recent SyntaxFest venue. Over the years, these venues have provided a central forum for high-quality research on treebanks, syntactic theory, syntax-semantics interface, structured meaning representations, and annotated linguistic resources. With record participation in recent years, they demonstrate the vitality and relevance of this line of work. The Workshop on Structured Linguistic Data is conceived as both a continuation of this tradition and an adaptation to the new realities of an LLM-dominated research landscape. The workshop brings together researchers from these overlapping traditions to advance methods, resources, and practices for integrating structured linguistic data into the LLM era.

The papers collected in this volume reflect a vibrant and multifaceted engagement with these issues, demonstrating that structured representations are not a relic of the past, but an essential component of robust, interpretable, and theoretically grounded natural language processing. A recurring theme across the program is the need to rigorously capture the syntax-semantics interface and complex event dynamics. For instance, contributions to the workshop explore how abstract meaning representations can be extended to handle quantification, and how subevent structures serve as vital predictors for entity identity changes in procedural texts. Such approaches underscore the necessity of deep lexical semantic analysis for tracking states and understanding discourse well beyond surface-level pattern matching.

Equally prominent is the focus on the creation, curation, and analysis of specialized corpora and lexicons. The accepted papers highlight the immense linguistic diversity and complexity that foundational models still struggle to autonomously master. Researchers present novel findings on human label variation in coreference, the compilation of temporally faithful corpora for diachronic NLP, and domain-specific challenges, such as those found in specialized German sermons. Furthermore, there is a strong emphasis on word-internal structures and morphological phenomena, ranging from large-scale segmentation across dozens of languages to the intricacies of linking elements in German nominal compounding and non-concatenative morphology in Tashlhiyt Berber. By modeling these regularities, including the design of datasets for regular metaphor, the community continues to provide the high-fidelity data necessary to push computational linguistics forward.

The direct interplay between generative models and structured data forms another critical pillar of this year's proceedings. Several authors tackle the evaluation and enhancement of LLMs through structured lenses, whether by comparing natural and synthetic structured data regarding verb alternations, or by exploring corpus-grounded agentic LLMs for multilingual grammatical analysis. Additionally, methods for improving language models for lexicographic question answering and employing syntax as a semantic pivot in machine translation showcase practical pathways for injecting rich linguistic knowledge into contemporary pipelines.

Finally, methodological innovations in how we store, query, and utilize structured linguistic data continue to evolve. The program features exciting work on fast and flexible example-based treebank search using vector symbolic architectures, offering fresh mathematical and computational perspectives on representing and retrieving complex structures. Together with the insights from our keynote speakers, Shira Wein and Naiara Perez, the research presented in these proceedings charts a clear and exciting course. It is our hope that this volume not only documents the remarkable work being done at this critical intersection, but also inspires continued theoretical and practical development across the community.

The number of submissions was 30 which shows the relevance of the topic. From these, 21 papers were accepted – 7 for oral presentation and 14 as posters. The acceptance rate is thus 70 % and it indicates the high quality of the submissions.

The Organizers

Acknowledgements

We would like to thank our invited speakers Naiara Perez and Shira Wein for their stimulating keynotes and for their willingness to cover their own accommodations and travel expenses. Also, we are grateful to have had Michał Ulewicz (IBM) as secondary reviewer. We are also indebted for the guidance provided by the LREC workshop committee that made our preparations a smooth process.

Organizing Committee

Jan Hajič (Prague University, Czech Republic)
Erhard Hinrichs (Tübingen University, Germany)
Sandra Kübler (Indiana University, USA)
Joakim Nivre (Uppsala University, Sweden)
Petya Osenova (Sofia University and IICT-BAS, Bulgaria)
James Pustejovsky (Brandeis University, USA)

Program Committee

Meriem Beloucif (Uppsala University, Sweden)
Bernd Bohnet (Google, London, UK)
Johan Bos (University of Groningen, The Netherlands)
Gosse Bouma (University of Groningen, The Netherlands)
Çağrı Çöltekin (Tübingen University, Germany)
Daniel Dakota (Leidos, Indiana University, USA)
Miryam de Lhoneux (KU Leuven, Belgium)
Stephanie Dipper (Bochum University, Germany)
Katrin Erk (University of Massachusetts at Amherst, USA)
Kilian Evang (Düsseldorf University, Germany)
Kim Gerdes (Université Paris-Saclay, France)
Eva Hajičová (Charles University, Czech Republic)
Yoshihiko Hayashi (Waseda University, Japan)
Shu-Kai Hsieh (National Taiwan University, Taiwan)
Laura Kallmeyer (Düsseldorf University, Germany)
Ekaterina Kochmar (Mohamed Bin Zayed University of AI, Abu Dhabi)
Marie-Pauline Krielke (Saarland University, Germany)
Lori Levin (Carnegie Mellon University, USA)
Wolfgang Menzel (Hamburg University, Germany)
Marie Mikulová (Charles University, Czech Republic)
Kaili Müürisep (University of Tartu, Estonia)
John Nerbonne (Freiburg University, Germany)
Colleen Alena O'Brien (Palacky University, Olomouc, Czech Republic)
Stephan Oepen (Oslo University, Norway)
Lilja Øvrelid (University of Oslo, Norway)
Barbara Plank (LMU Munich, Germany)
Agata Savary (Université Paris-Saclay, France)
Kiril Simov (IICT, Bulgarian Academy of Sciences, Bulgaria)
Sanghoun Song (Korea University, Korea)
Sara Stymne (Uppsala University, Sweden)
Rob van der Goot (IT University of Copenhagen, Denmark)
Leonie Weissweiler (Uppsala University, Sweden)
Shuly Wintner (Haifa University, Israel)
Alina Wróblewska (Polish Academy of Sciences, Warsaw, Poland)
Nianwen Xue (Brandeis University, USA)
Daniel Zeman (Charles University, Czech Republic)
Heike Zinsmeister (Hamburg University, Germany)

Keynote Speakers

Naiara Perez (University of the Basque Country)
Shira Wein (Amherst College)

Table of Contents

Workshop Program

Monday, May 11, 2026

09:00–09:10 **Opening**
Room: 3
Chairs: Erhard Hinrichs, Joakim Nivre, Petya Osenova

09:10–10:00 **Keynote 1: Shira B. Wein, Amherst College**
Room: 3
Chair: James Pustejovsky

10:00–10:30 **Session 1**
Room: 3
Chair: James Pustejovsky

10:00–10:30 *Degrees of Subjectivity and Their Repercussions in Conversation. The View from Online Interactions*
Gonzalo Freijedo Aduna, Anastasia Giannakidou and Alda Mari

10:30–11:00 **Coffee Break 1**

11:00–13:00 **Session 2**
Room: Room 3
Chair: Petya Osenova

11:00–11:30 *Constraints on Linking Element Choice in German Nominal Compounding: A Large-Scale Corpus Study*
Maksim Shmalts

11:30–12:00 *Comparing Natural and Synthetic Structured Data: A Study of the Passive Verb Alternation in French and Italian*
Giuseppe Samo and Paola Merlo

12:00–12:30 *Subevent Structure as a Predictor of Entity Identity Change in Procedural Text*
Kyeongmin Rim and James Pustejovsky

12:30–13:00 *Fast and Flexible Example-Based Treebank Search with Vector Symbolic Architectures*
Adam Roussel

Monday, May 11, 2026 (continued)

13:00–14:00 **Lunch Break**

14:00–15:10 **Poster Session**
Chair: Joakim Nivre

14:00–15:10 *Modular Neural Machine Translation with a Semantic Pivot - Pilot Study Using AMR*
Wenyang Gao, Yaxuan Li, Yunxin Bao, Shulin Huang and Yue Zhang

14:00–15:10 *Gutenberg+: A More Temporally Faithful Corpus for Diachronic NLP*
Leon Hammerla and Alexander Mehler

14:00–15:10 *Automatic Lemmatisation for Norwegian*
Ahmet Yildirim, Kristin Hagen and Dag Haug

14:00–15:10 *Quantification in Abstract Meaning Representation*
Kiyong Lee and Chongwon Park

14:00–15:10 *Improving Slovene Language Models for Lexicographic Question Answering through Continued Pretraining and Instruction Fine-Tuning*
Timotej Knez and Slavko Zitnik

14:00–15:10 *Structured Partial Predictability in Non-Concatenative Morphology: The Case of Tashlhiyt Berber*
John Alderete and Hamza Sellami

14:00–15:10 *Towards Corpus-Grounded Agentic LLMs for Multilingual Grammatical Analysis*
Matej Klemen, Tjaša Arčon, Luka Terčon, Marko Robnik-Sikonja and Kaja Dobrovoljc

14:00–15:10 *The L2 Network: A CEFR-Aligned Knowledge Graph for Grammar Domain Modeling*
Luisa Ribeiro-Flucht and Xiaobin Chen

14:00–15:10 *Paraphrase Acquisition via Bilingual Pivoting Based on Neural Word Alignment*
Risa Kondo, Seiji Sugiyama, Tomoyuki Kajiwara and Takashi Ninomiya

Monday, May 11, 2026 (continued)

14:00–15:10 *Using syntax for the semantic representation of sentences*
Iskandar Boucharenc, Eve Sauvage, Thomas Gerald, Julien Tourille, Sabrina Campano, Cyril Grouin and Sophie Rosset

14:00–15:10 *Modeling Word-Internal Structures: Morphological Segmentation Across 58 Languages*
Vojtěch John, Zdeněk Žabokrtský and Benjamin Reeves

14:00–15:10 *DiNoS: Creating a Data-Driven German Noun Phrase Lexicon from Universal Dependencies*
Jacob Lee Suchardt and Ronja Laarmann-Quante

14:00–15:10 *Figurative, Polysemous, Conventional: Designing a Dataset of Regular Metaphor*
Anna Temerko, Pablo Gamallo and Marcos Garcia

14:00–15:10 *Domain-Specific Considerations in the Preparation of Specialized Corpora: A Case Study on a Corpus of German Sermons*
Cora Haiber, Adam Roussel and Stefanie Dipper

15:10–16:00 **Keynote 2: Naiara Perez, University of the Basque Country**
Room: 3
Chair: Petya Osenova

16:00–16:30 **Coffee Break 2**

16:30–17:30 **Session 3**
Room: 3
Chair: Jan Hajič

16:30–17:00 *Semantic, Syntactic, Lexical: What Makes QA Augmentation Work in Limited Quantity?*
Benedictus Kent Rachmat, Thomas Gerald, Takuya Nakamura, Zheng Zhang and Cyril Grouin

17:00–17:30 *Introducing corpora Hlava Cor and Hlava AD: Human Label Variation in Coreference and Discourse Relations*
Anna Nedoluzhko, Sarka Zikanova, Jiri Mirovsky, Milan Straka and Eva Hajicova

Monday, May 11, 2026 (continued)

17:30–18:00 **Closing Discussion**
Room: 3
Chairs: Jan Hajič, Erhard Hinrichs, Joakim Nivre, Petya Osenova, James Pustejovsky

Keynote Speech
When Does Linguistics Help? Downstream Utility of Linguistic Frameworks in the Age of LLMs

Shira Wein
Amherst College

Abstract

Simply by training on massive amounts of data, large language models (LLMs) can produce highly fluent text across many languages. While structured linguistic resources—such as treebanks—and theoretically grounded linguistic frameworks have long played a central role in advancing NLP, the rise of LLMs raises important questions about their continued utility in this new paradigm. In this talk, I will present several applications of linguistic frameworks, including visual question answering, machine translation for low-resource languages, and classifier model interpretability. These applications demonstrate the role of linguistics in the age of LLMs. I will conclude by outlining promising directions for future research, highlighting open areas in NLP where linguistic frameworks can play a critical role.

Keynote Speech
Building and Evaluating Instruction-Following LLMs for Low-Resource Languages

Naiara Perez
University of the Basque Country

Abstract

The development of large language models (LLMs) for low-resource languages has progressed from a largely theoretical challenge to an active area of empirical research. This talk uses Basque as a case study to examine two open problems that extend to low-resource languages more broadly: how to build instruction-following LLMs for languages with limited resources, and how to evaluate them. Building on Latxa (a family of open Basque LLMs), we present two complementary approaches to instruction-tuning, fine-tuning and model merging, which we validate through a combination of standard benchmarks, verifiable instruction following, and a community evaluation arena. The second part of the talk frames evaluation itself as an open research problem, presenting ongoing work on cross-lingual benchmarking and human evaluation methodology, and discussing what the latter reveals about the challenges of assessing linguistic quality in low-resource settings.

Degrees of Subjectivity and their Repercussions in Conversation
The View from Online Interactions

Gonzalo Freijedo Aduna[1], **Anastasia Giannakidou**[2], **Alda Mari**[1]
[1]Institut Jean Nicod, CNRS/ENS/EHESS/PSL University
[2]The University of Chicago
gonzalo.freijedo.aduna@ens.psl.eu, giannaki@uchicago.edu, alda.mari@ens.psl.eu

Abstract

We present the **Annotated Reddit Conversation Corpus (ARCC)**, an English-language dataset of online discussions annotated for *Speech Acts* and *Functional Dependence Relations*, designed to investigate how varying degrees of subjectivity influence conversational dynamics and interaction patterns. At the speech act level, we distinguish factual from opinion statements and further classify opinions along a five-degree scale of subjectivity. Functional Dependence Relations capture how segments relate to preceding ones. Analyses show that opinion-discussion contexts feature frequent inter-subjective opinions eliciting explicit agreement and disagreement, while information-exchange contexts exhibit less subjective opinions with responses like answers or requests for clarification. We further demonstrate that a transformer model can predict the subjectivity scale with promising performance. The corpus and annotation guidelines are made available to support future research on opinion expression and automated dialogue analysis.

Keywords: Pragmatic Annotation, Speech Acts, Functional Dependence Relations, Subjectivity

1. Introduction

Language is used not only to describe the world but also to express perspectives, evaluations, and normative stances (Portner, 2009). In everyday conversation, which today also includes online interaction, speakers constantly alternate between statements that aim to represent how the world objectively is and statements whose truth depends, wholly or partially, on an evaluation parameter. The latter are therefore subjective to varying degrees.

Our goal is to examine the pragmatic impact of subjectivity in online discourse, specifically how and to what extent subjective statements (classified along a graded subjectivity scale) influence the trajectory of interactions.

Formal models of dialogue have long emphasized that different conversational moves, particularly those carrying distinct illocutionary force, constrain the range of possible follow-up moves (Farkas and Bruce, 2010; Ginzburg, 2012; Krifka, 2022). On the other hand, research in NLP and computational linguistics has addressed subjectivity as a gradable phenomenon. However, to the best of our knowledge, the interaction between these two lines of inquiry (namely, how graded subjectivity affects the dynamics of conversational structure) has not yet been systematically investigated.

To address this, we introduce a corpus of Reddit conversations (**ARCC**, Freijedo Aduna et al., 2026) drawn from different Reddit communities[1]. Comments were segmented into minimal communicative units (or functional segments), each of which was annotated at two levels:

1. **Speech act annotations**, identifying the illocutionary force of each segment. At this level, we distinguish between statements and questions that provide or request factual information, and those that concern opinions. The latter were classified in a five-degree scale of subjectivity. This distinction enables us to differentiate between information-exchange contexts (primarily governed by the former) and opinion-discussion contexts.

2. **Functional Dependence Relation annotations**, capturing the communicative function a segment fulfills in relation to the segment to which it is linked. This layer of annotation allows us to examine whether, and how, the responses or follow-up contributions vary across different conversational contexts.

Our distinction between factual and opinion statements builds on a twofold distinction proposed by Giannakidou and Mari (2021a). The first dimension differentiates between *veridical* and *nonveridical* commitment, while the second contrasts *exogenous* and *endogenous* evidence. As justified in Section 2, we classify both nonveridical statements and statements grounded in endogenous evidence as opinion statements.

Extending this approach, we introduce a five-degree scale of subjectivity based on the type of evidence supporting the claim. The scale spans from maximally subjective judgments, such as predicates of personal taste, where the relevant evidence is personal, unshareable experience, to comparatively less subjective epistemic modals as (1),

[1] https://github.com/gFreijedo/arcc2026-subjectivity.

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 1–13
May 11, 2026.

whose truth conditions ultimately depend on publicly accessible evidence.

(1) Speaker (Seeing people arrive with wet umbrellas): It must be raining.

We experimented with a transformer model (`RoBERTa-base`) in order to test its capacity to learn the scale of subjectivity, obtaining relatively good results.

By operationalizing these distinctions through corpus annotation, we provide an empirical framework for investigating how degrees of subjectivity shape discourse structure and condition the types of reactions that emerge in different communicative contexts.

Our analyses show that opinion-discussion contexts are characterized by high frequencies of intersubjective opinions accompanied by explicit agreement and disagreement, whereas information-exchange contexts feature less subjective opinions and responses such as answers, continuers, or requests for clarification.

In this perspective, a natural downstream application of our work is the improved identification of opinion- versus fact-oriented statements in dialogue, taking into account both their degree of subjectivity and their interactional profile. In particular, given a sufficiently large corpus, it would be possible to model, for each segment, the distribution of responses it is likely to elicit (e.g., agreement, disagreement, elaboration, or clarification requests). Such predicted interactional patterns could then be used as additional features for distinguishing between fact- and opinion-oriented statements, as well as for refining the estimation of subjectivity degrees. More generally, this approach opens the way to integrating conversational dynamics into tasks such as stance detection, argument mining, and the analysis of deliberative or non-cooperative dialogue.

The paper is organized as follows. Section 2 presents our theoretically grounded binary distinction between fact and opinion statements, along with a finer-grained classification of opinions on a scale of subjectivity. Section 3 reviews related work. Section 4 describes the corpus collection and annotation procedure. Before presenting the annotation results in Section 6, we report a transformer-based evaluation of learnability in Section 5. Finally, Section 7 concludes the paper.

2. Conceptual definitions

In this section, we discuss the theoretical foundation of both the binary distinction between fact- and opinion-statements and the gradation of subjectivity that we propose within the latter.

	Evidence			
	Exogenous		Endogenous	
	Total	Partial	Total	Partial
Veridical commitment	Fact	----	Opinion	----
Nonveridical commitment	----	Opinion	----	Opinion

Table 1: Distinction between fact-statements and opinion-statements based on the *type* and the *amount* of evidence.

2.1. Fact- vs. Opinion-statements

For the binary distinction between fact-statements and opinion-statements, we rely on two notions issued from the semantic literature: *veridical commitment* (Giannakidou and Mari, 2021a,b) and *type* of evidence (Nuyts, 2001; Karttunen and Zaenen, 2005; Aikhenvald and Aikhenvald, 2006; Giannakidou and Mari, 2021a,b).

Taking a *veridical commitment* means that, by uttering a plain assertion such as (2) in a cooperative conversation, the speaker is fully committed to the truth of the proposition expressed by the utterance. In contrast, by using epistemically modalized sentences like (3), speakers signal a *nonveridical commitment*, which means that they leave open the possibility of the complement being false.

(2) Yeasts metabolize carbohydrates into carbon dioxide gas and ethanol (alcohol).[2]

(3) That may give you enough room to get the outlet out.

When following rationality and cooperative principles (Grice, 1975), speakers shape their contributions according to the evidence available. Thus, in (2), the speaker signals sufficient evidence on the topic, whereas in (3), the speaker explicitly indicates a lack thereof. Accordingly, we treat sentences like (3) as opinion-statements, in which the speaker's choice of words signals insufficient or partial evidence for their claims. That is, in our classification, ***nonveridical* statements are treated as opinion statements**. In this respect, we depart from Nuyts (2001), who argues that epistemic modals are not *necessarily* subjective.

However, this does not imply that all opinion-statements are *nonveridical*. On the contrary, speakers are, more often than not, fully committed to their opinions. To determine whether a fully committed speaker is asserting a fact or expressing an opinion, it is necessary to consider the *type* of the evidence on which their assertion can be based.

[2] All of the following examples are drawn from our corpus, except for those marked with an asterisk (*), which are constructed for illustrative purposes.

Following Giannakidou and Mari (2021a), we distinguish between **exogenous** and **endogenous** evidence. Exogenous evidence refers to informational content, that is, "a body of knowledge that includes 'facts'". In hearing (2), for instance, an interlocutor would assume that this kind of evidence is available. Endogenous evidence, by contrast, refers to emotive or affective content, including "desires and hopes [...], as well as political, ideological, religious, or aesthetic beliefs" (Giannakidou and Mari, 2021a). In uttering (4), the speaker is expressing an opinion based on his sociological or political beliefs.

(4) Abortion is Healthcare

We consider that, *alongside nonveridical statements*, **statements based on endogenous evidence are opinion-statements**. Figure 1 summarizes these considerations[3].

2.2. Degrees of subjectivity

However, as noted in the Introduction, expressions of opinion vary in their degree of subjectivity.

The epistemic modals discussed in the previous section are less subjective than, for instance, predicates of personal taste. The latter express highly subjective judgments, grounded solely in the speaker's personal experience. For this reason, the linguistic literature characterizes them as giving rise to cases of *faultless disagreement* (among numerous works, see MacFarlane 2005; Lasersohn 2005; Stephenson 2007; Sæbø 2009; Kennedy 2013; Umbach 2016; Coppock 2018).

We propose that expressions involving communal and moral values occupy an intermediate position. Although they are still based on endogenous evidence, that is, they reflect personal beliefs, they are intrinsically *inter-subjective*. Community and normative values are, by their very nature, oriented toward regulating social behavior and guiding interactions among individuals. In expressing opinions about such matters, speakers necessarily take others into account[4].

[3]The notion of partial endogenous evidence is more difficult to characterize, since speakers typically have full introspective access to their own beliefs. A more accurate formulation might be that of *conflicting* endogenous evidence, as in cases such as "Perhaps I should speak out, even if it's uncomfortable(*)". As this distinction is not central to our annotation scheme, we leave a more precise account for future work.

[4]We remain neutral with respect to the philosophical debate between moral realism and moral relativism, focusing instead on how moral evaluations function as inter-subjectively recognized commitments within discourse, in line with John Searle's account of institutional reality (Searle, 1995).

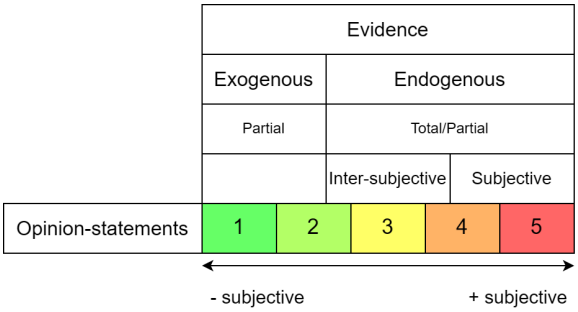


Table 2: Distinction between subjective and inter-subjective endogenous evidence and grades of subjectivity of opinion-statements.

So far, we have distinguished three levels of subjectivity, based on the type and amount of available evidence: the first corresponds to the facts of the world, or exogenous evidence when partial; the second to inter-subjective endogenous evidence; and the third to purely subjective endogenous evidence. However, as we will illustrate with examples from our corpus in Section 4, we also take into account cases in which multiple types of evidence are intertwined. In some instances, for example, an utterance simultaneously invokes personal taste and moral evaluation. Consider the following example:

(5) It would be fun if it weren't financed with dirty money*.

In this case, the speaker expresses a judgment of personal taste (it would be fun), while at the same time introducing a moral evaluation (financed with dirty money). The overall interpretation thus combines highly subjective experiential evidence with inter-subjective normative considerations.

There are also cases in which speakers evaluate real-world events on the basis of personal beliefs, thereby combining exogenous evidence with endogenous, inter-subjective evidence, as in (6). In such instances, reference is made to publicly accessible facts or events (in the example, historic events), but the assessment itself is filtered through normative or value-laden commitments. The resulting judgment is thus anchored in the external world while simultaneously reflecting the speaker's evaluative stance.

(6) That's probably the best outcome an English king could have hoped for

These considerations ultimately lead us to propose a five-degree scale of subjectivity (Figure 2). The scale ranges from (1-WORLD) statements with minimal subjectivity, whose truth can, in principle, be settled by facts of the world; (2-WORLD+VALUES) statements combining objective-world conditions and inter-subjective values; (3-VALUES) statements

based solely on inter-subjective or community values; (4-TASTE+VALUES) statements combining inter-subjective values and personal taste; and (5-TASTE) statements evaluated relative to an individual parameter, such as predicates of personal taste, representing the maximal level of subjectivity.

3. Related Work

In recent years, substantial effort within computational linguistics and NLP communities has been devoted to distinguishing factual from opinionated or subjective statements, often as a preparatory task for opinion mining and sentiment analysis. For a broad overview of frameworks, techniques, and applications in sentiment analysis, see Chandan and Mandal (2025). Although this factual–subjective distinction is frequently motivated by sentiment analysis tasks, our focus here is different: we concentrate on corpus annotation projects that model textual and conversational subjectivity, as well as recent work closely related to our own.

A seminal contribution to the study of subjectivity in natural language is Wiebe et al. (2005), who propose an annotation scheme for expressions of opinions and emotions at the phrase or clause level in news articles. Their framework provides a fine-grained representation of subjective content. Our work extends this line of research to conversational data, examining how expressions of opinion in Reddit comments shape and constrain subsequent contributions.

More recently, Antici et al. (2024) introduced a corpus of sentences extracted from news articles and classified as either subjective or objective, following highly prescriptive annotation guidelines. By contrast, Savinova and Moscoso Del Prado (2023) adopt the opposite approach, relying on naïve annotators who are provided only with brief definitions of objective and subjective statements. A key contribution of their work is to move beyond a purely binary distinction and to treat subjectivity as a gradable scale. In our study, we combine insights from both approaches: we provide theoretically informed annotation guidelines while modeling subjectivity expression as an ordinal five-level scale.

Another approach to the classification of subjective statements is proposed by Benamara et al. (2011), who introduce a taxonomy of elementary discourse units (EDUs) for the automatic detection of subjectivity in discourse (in this case, online film reviews). They distinguish four classes: (1) explicit subjectivity, identified using a subjectivity lexicon; (2) implicit subjectivity, whose interpretation must be resolved from context; (3) non-evaluative subjectivity, capturing propositional attitudes treated as separate EDUs from their complements; and (4) non-subjective EDUs. While the present study focuses on linguistically explicit expressions of subjectivity, we acknowledge the importance of accounting for their implicit realizations, which we leave for future work.

Closely related to our project, the influential FactBank corpus (Saurí and Pustejovsky, 2009) provides an annotation framework for capturing degrees of factuality in text, labeling events as true, possible, or counterfactual. Although FactBank, like the previously mentioned resources, is not based on conversational data, its emphasis on fine-grained factuality and on the interaction between modality and evidence informs our treatment of opinion statements and supports our distinction between factual and subjective contributions in dialogue.

Considering conversational corpora, a key annotation project is that of Stolcke et al. (2000), who focus on dialogue act modeling for the automatic tagging of conversational speech. Their classification includes a distinction between factual and opinion statements, grounded in pragmatic considerations. Building on this idea, we define finer-grained opinion categories to capture more nuanced conversational phenomena, informed by recent developments in semantic and pragmatic theory.

The STAC corpus (Asher et al., 2016) contains multi-party chat dialogues annotated for rhetorical relations and dialogue acts using SDRT (Asher and Lascarides, 2003). The ANNODIS project (Péry-Woodley et al., 2011) provides French online dialogues annotated for rhetorical relations and dialogue acts. Both serve as references for multi-level annotation of online conversations.

Similarly, the ISO standard 24617-2 (Bunt et al., 2012, 2020) offers a highly detailed framework for dialog act annotation. While this level of precision is valuable, it is often too complex for our purposes. We therefore adopt a simpler scheme, but borrow from ISO 24617-2 the notion of functional dependence relation to characterize how one segment relates to another.

Finally, a note on discourse parsing. A major challenge in the annotation process was reconstructing reply-to chains by linking each segment to the segment it responds to. Automatic discourse parsing has addressed related tasks through sequential modeling of discourse structure (Hernault et al., 2010), joint modeling of speech acts and discourse relations (Joty and Mohiuddin, 2018), and topic segmentation with relation labeling (Joty et al., 2013). In future work, we plan to explore similar approaches in order to enable the expansion of the corpus.

Subreddit	Posts	Comments	Segments
Askpolitics	3	40	145
DIY	8	78	254
askscience	11	111	448
changemyview	7	82	458
prochoice	5	99	315
todayilearned	13	159	310
Total	**47**	**569**	**1930**

Table 3: Corpus Statistics by Subreddit

4. Corpus and annotation procedure

In this section, we describe the collection of our corpus, the segmentation of the comments, the reorganization of the reply chains, and our two-level annotation scheme.

4.1. Corpus collection and segmentation

We collected the data by scraping six Reddit communities (subreddits): *AskPolitics*, *askscience*, *changemyview*, *DoItYourself*, *prochoice*, and *todayilearned*. These subreddits were selected to represent a range of communicative contexts.

Since Reddit comments often contain multiple communicative moves within a single turn, we segmented each comment into smaller units to annotate the parts of the dialogue that fulfill distinct communicative functions, called *functional segments* in ISO 24617-2 (Bunt et al., 2012). To achieve this, we first used *Stanza* (Qi et al., 2020) for tokenization, part-of-speech tagging, dependency parsing, and sentence segmentation. We then applied additional rule-based procedures to obtain finer-grained utterance units. This process consisted in detecting discourse markers (e.g., *but*, *because*, *although*) and multi-word connectors (e.g., *even though*, *so that*), while also relying on syntactic cues such as the presence of both subject and predicate to prevent the improper splitting of coordinated verb phrases. Each segment was assigned a unique identifier based on the original comment ID. Minor segmentation errors were subsequently corrected during the manual annotation phase. The resulting corpus comprises a total of 47 posts, 569 comments, and 1,930 segments; a breakdown by subreddit is provided in Table 3.

We subsequently restructured the reply-to chain so that each segment is explicitly connected to the precise segment it addresses. In the vast majority of cases, a segment is linked to the immediately preceding segment within the same comment. Two main exceptions are: (1) the first segment of a comment, which is typically linked to the final segment of the comment it replies to, and (2) segments that explicitly quote or directly address a non-adjacent segment. In (7), for instance, segment $B1$ is linked to $A1$ and $B2$, to $A2$.

(7) A: [What year did you graduate]$_{A1}$ [and how was it?]$_{A2}$
B: [2029,]$_{B1}$ [it was awesome]$_{B2}$

4.2. Annotation Scheme

The annotation was carried out at two levels: *Speech Acts* and *Functional Dependence Relations*.

By *Speech Acts*, we refer to the illocutionary force of an utterance in the sense of Searle (1969), i.e., the type of communicative act intended by the speaker when producing the utterance (e.g., asserting, questioning, advising). Each annotated speech act corresponds to a *functional segment* as defined in Bunt et al. (2020).

Functional Dependence Relations capture the communicative function of a segment with respect to a preceding segment. The notion is adopted from Bunt et al. (2020), who define it as a "relation between a given dialogue act and a preceding dialogue act on which the semantic content of the given dialogue act depends due to its communicative function". In this sense, Functional Dependence Relations are closely related to the linguistic notion of *Rhetorical Relations* in SDRT (Asher and Lascarides, 2003), as both describe structured dependencies between discourse units. However, Rhetorical Relations are more general, since they also account for coherence relations between segments within a discourse more broadly, not limited to dialogical interactions between speakers

To ensure the reliability of the annotation scheme, the first 500 segments were independently annotated by two PhD students. This resulted in an inter-annotator agreement of **Cohen's Kappa = 0.78**, indicating substantial agreement. The remaining 1,430 segments were annotated according to the established guidelines to complete the GOLD standard. The fine-grained Subjectivity Scale for opinion statements (ASEESSMENT in our annotation scheme) was annotated by a single annotator from the initial annotation team. Since it was not possible to compute inter-annotator agreement for this layer, we later evaluate the operationalizability of the scale through a supervised learning experiment (see Section 5 below), testing whether the distinctions can be reliably learned by a transformer-based model.

Speech Acts. We distinguish the following speech act categories and refer the reader to Appendix A for illustrative examples drawn from our corpus:

- ASSESSMENT (ASSESS): statements describing how the world is or should be from the speaker's perspective.

- QUESTION (OPINION) (Q_OPIN): questions seeking an assessment as a response.
- ASSERTION (ASSERT): statements presenting facts without linguistically marked subjectivity.
- QUESTION (FACT) (Q_FACT): questions seeking factual information[5].
- EXPRESSIVE (EXPR): expressions of emotion, gratitude, greetings, irony, or humor.
- ADVISE (ADV): utterances intended to guide or influence the hearer's actions (typically, though not necessarily, in imperative form).
- OTHER (OTH): utterances that do not fall into the previous categories.

Subjectivity Scale for Assessments. ASSESSMENT instances were further subdivided into five subclasses reflecting increasing degrees of subjectivity based on the considerations presented in Section 2.2. In the examples below, the relevant linguistic cues are underlined:

- WORLD (WOR): opinions about factual matters (e.g., "That may give you enough room to get the outlet out").
- WORLD + VALUES (V/W): opinions about factual matters involving evaluation based on communal or inter-subjective values (e.g., "Putin's Russia is a mafia state").
- VALUES (VAL): opinions grounded in communal values, worldviews, or moral standpoints (e.g., "drones should be banned for civilian use").
- TASTE + VALUES (T/V): opinions based on personal taste intertwined with communal values (e.g., "but cool idea nonetheless").
- TASTE (TST): opinions grounded purely in personal taste predicates (e.g., "They still look as good as the day they went outside.").

Functional Dependence Relations. We annotate the relation between each segment and the segment it is linked to using the following categories, and refer the reader to Appendix B for illustrative examples from our corpus:

- AGREEMENT (AGR): explicit agreement with the linked segment.
- DISAGREEMENT (DIS): explicit disagreement with the linked segment.
- ANSWER (ANS): a statement responding to a linked question.
- REQUEST FOR CLARIFICATION (REQ): a question or statement seeking further specification of the linked segment.
- CONTINUATION–IN (CONT_IN): a segment linked to the immediately preceding segment within the same comment.
- CONTINUATION–EX (CONT_EX): a segment linked to a segment in another comment without signaling explicit agreement or disagreement.
- OFF-TOPIC (OFF): a segment that diverges from the topic of the linked segment.

[5]Both question types were additionally marked for *bias*, since interlocutors may agree or disagree with the presupposed or implied stance conveyed by a question.

5. Model-Based Evaluation of the Scale

To assess the internal coherence and operationalizability of our five-degree subjectivity scale, we first extracted all ASSESSMENT segments from the corpus. These opinion-bearing segments were then used as input to a supervised learning experiment with `RoBERTa-base` (Liu et al., 2019), following the general approach of Savinova and Moscoso Del Prado (2023). The five labels (WOR, V/W, VAL, T/V, TST) were mapped to numerical values from 1 to 5 and then normalized to the range 0–1 for regression. The model was trained with mean squared error loss, a small learning rate (1e-5), and early stopping.

Given the relatively small size of our dataset (835 English segments), we perform stratified 5-fold cross-validation to ensure stable and reliable performance. The model shows consistent results across folds, with *Mean Absolute Error* (MAE) = 0.1568 $\pm$ 0.0108 and *Root Mean Squared Error* (RMSE) = 0.2104 $\pm$ 0.0134, along with moderate-to-strong correlations (*Pearson's* r = 0.6689 $\pm$ 0.0475, *Spearman's* ρ = 0.6379 $\pm$ 0.0493), indicating stable performance across splits.

We then train a final model on an 80% training split and evaluate it on a held-out 20% test set. On the held-out data, the model achieves MAE = 0.1000, RMSE = 0.1465, Pearson r = 0.8563, and Spearman ρ = 0.8298, indicating strong linear and ordinal correspondence with the gold labels.

Figure 1 shows the linear regression between gold and predicted values on the test set ($R^2 = 0.733$), indicating that the model's predictions closely track the annotated subjectivity scores. The plot also reveals a slight bias at the extremes: for the least subjective category (1), the model predicts higher subjectivity than gold, while for the most subjective category (5), it predicts lower subjectivity.

Figure 2 presents the confusion matrix after rounding predictions to the nearest category, with 74.4% exact accuracy. Most misclassifications occur between adjacent categories, while confusions across distant categories are rare, supporting the

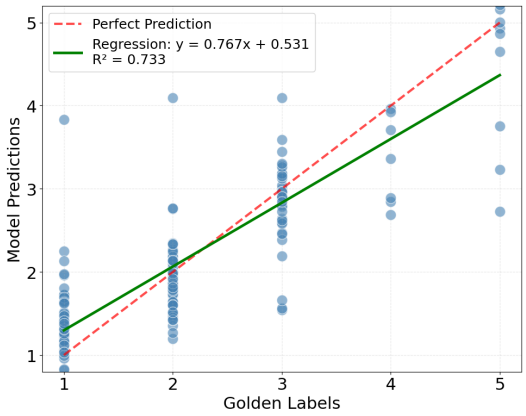


Figure 1: Linear regression between gold and predicted subjectivity scores on the held-out test set ($R^2 = 0.733$).

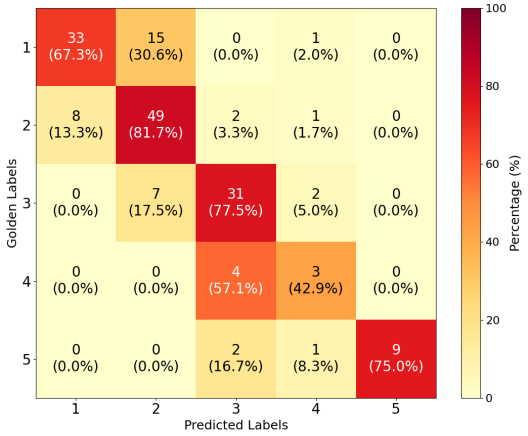


Figure 2: Confusion matrix showing exact-match prediction accuracy of 74.4%, computed across all samples and weighted by class frequencies.

graded structure of the scale. The most notable misclassification is for gold label 4 (T/V), which the model predicts as 3 in 57% of cases, showing occasional underestimation of higher degrees of subjectivity.

This pattern is partly due to the small number of examples for the high-subjectivity categories in our corpus (T/V and TST), which makes predictions for these classes less reliable. Additional annotated data for these categories would be necessary to achieve more conclusive results across the full scale of subjectivity.

We manually inspected the model's predictions that deviate by more than one degree from the gold label in order to better understand systematic sources of error. Due to space constraints, we illustrate this analysis with a small number of representative examples.

In (8), the model appears to assign greater weight to the expression "it's odd", interpreting it as an evaluation grounded in personal taste. However, under our annotation scheme, the property being evaluated (the absence of information about the person) is a fact about the world. Therefore, the correct label is 2, as the judgment evaluates an external, world-related state of affairs.

(8) Its odd but there's practically nothing about this guy in a Google Search. (True label: 2 — Values + World; Predicted label: 4 — Taste + Values)

In (9), the model seems to focus on the category "American", potentially interpreting the statement as invoking socially shared values or norms. However, the relevant evaluative component is "feel fake", which in our framework constitutes fully endogenous evidence. As such, the correct label is 5, reflecting a judgment grounded in personal taste rather than externally anchored values.

(9) And most of the "American" profiles feel fake. (True label: 5 — Taste; Predicted label: 3 — Values)

Although missing the correct label by only one degree, (10) highlights an important limitation of our approach: the model does not incorporate conversational context, in particular the content of the segment being responded to. In such cases, the interpretation of the utterance depends entirely on its antecedent, and the appropriate label is inherited from the preceding segment. The model's inability to account for this dependency leads to systematic misclassification of context-dependent responses.

(10) Exactly! (True label: 3 — Values; Predicted label: 4 — Taste + Values)

It is important to note that this experiment does not replace inter-annotator agreement; rather, it provides computational evidence that the annotation scheme encodes a coherent and learnable dimension of subjectivity grounded in systematic linguistic cues. It is also worth noting that the results reported correspond to the best-performing run, as performance shows non-negligible variance across different data splits. Future work could address this instability through data augmentation or ensemble techniques. The best-trained model is available in our repository.

6. Results

In this section, we present the main findings of our annotation, considering both the binary distinction between statements of fact and opinion, as well as the finer-grained degrees of subjectivity.

Subreddit	ASSESS	Q_OPIN	ASSERT	Q_FACT	EXPR	ADV	OTH	TOTAL
AskPolitics	84 (57.9%)	14 (9.7%)	38 (26.2%)	0 (0.0%)	4 (2.8%)	4 (2.8%)	1 (0.7%)	145
changemyview	343 (74.9%)	22 (4.8%)	63 (13.8%)	6 (1.3%)	12 (2.6%)	6 (1.3%)	6 (1.3%)	458
prochoice	164 (52.1%)	26 (8.3%)	56 (17.8%)	3 (1.0%)	51 (16.2%)	10 (3.2%)	5 (1.6%)	315
askscience	84 (18.8%)	4 (0.9%)	278 (62.1%)	61 (13.6%)	11 (2.5%)	10 (2.2%)	0 (0.0%)	448
DoItYourself	65 (25.6%)	4 (1.6%)	84 (33.1%)	26 (10.2%)	25 (9.8%)	46 (18.1%)	4 (1.6%)	254
todayilearned	96 (31.0%)	8 (2.6%)	144 (46.5%)	22 (7.1%)	28 (9.0%)	8 (2.6%)	4 (1.3%)	310
TOTAL	**836**	**78**	**663**	**118**	**131**	**84**	**20**	**1930**

Table 4: Distribution of speech acts across subreddits (counts with row-wise percentages).

6.1. Binary Classification

We first examined the distribution of speech acts across the six subreddits in our corpus (Table 4). For analysis, we grouped the subreddits into two *Global Communicative Contexts* based on the expected type of interaction: *opinion-discussion* (*AskPolitics*, *changemyview*, *prochoice*) and *information-exchange* (*askscience*, *DoItYourself*, *todayilearned*). Figure 3 presents a heatmap of Pearson residuals, highlighting the under- and over-representation of each speech act category within each group relative to the expected distribution under independence. In Opinion-Discussion contexts, ASSESSMENT and QUESTION (OPINION) are over-represented, while ASSERTION and QUESTION (FACT) are under-represented. The opposite pattern is observed in Information-Exchange contexts, reflecting their focus on factual information rather than evaluative or opinion-based contributions.

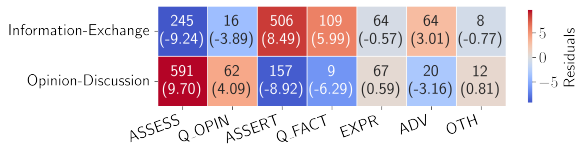


Figure 3: Heatmap of Pearson residuals showing the over- and under-representation of speech act categories across the two Global Communicative Contexts.

To examine how the two global communicative contexts shape conversational structure and dynamics, we then analyzed the distribution of Functional Dependence Relations in each context, shown in Figure 4. The results reveal distinct rhetorical profiles aligned with each context's communicative function.

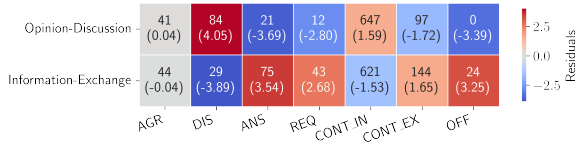


Figure 4: Heatmap of Pearson residuals for Functional Dependence Relations by global communicative context.

Opinion-discussion contexts show a strong over-representation of DISAGREEMENT (DIS) relations, reflecting the deliberative and argumentative nature of these communities where debate and negotiation of opposing viewpoints are central. In contrast, information-exchange contexts exhibit over-representation of ANSWER (ANS) and REQUEST FOR CLARIFICATION (REQ) relations, consistent with communities focused on collaborative knowledge sharing and uncertainty resolution.

We can also analyze patterns of Functional Dependence Relations by focusing on the immediate, local context, examining the types of relations that segments labeled as ASSERTION and ASSESSMENT receive from segments in replying comments (i.e., from other speakers) without considering the broader communicative context. As shown in Figure 5, ASSESSMENT segments are more likely to receive responses that explicitly agree or disagree with their content, whereas ASSERTION segments tend to elicit requests for clarification or continuers, consistent with the trends observed in the global discourse context.

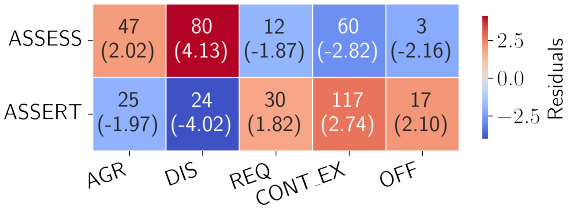


Figure 5: Heatmap of Pearson residuals for Functional Dependence Relations by local communicative context.

6.2. Five-Grade Subjectivity Classification

Moving beyond the binary fact/opinion distinction, we examine the distribution of opinions across our fine-grained five-grade subjectivity scale. This scale captures gradations of subjectivity, from opinions based on personal taste (TST) to opinions about world facts (WOR), with taste+value (T/V), values (VAL), and world facts+values (V/W) occupying intermediate positions.

Table 5 shows raw count and percentages in the whole corpus. Figure 6 shows the distribution of

SUB	Count	Proportion (%)
WOR	245	29.31
V/W	297	35.53
VAL	200	23.92
T/V	37	4.43
TST	57	6.82

Table 5: Distribution of Sub-Assessment Labels in the Corpus

these five opinion types across two global communicative contexts identified earlier.

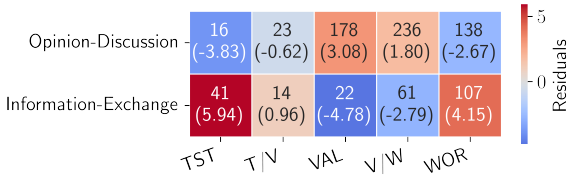


Figure 6: Heatmap of Pearson residuals for Sub-classes of ASSESSMENT by global communicative context.

In Information-Exchange contexts (*askscience*, *DoItYourself*, *todayilearned*), we observe over-representation of the most subjective category (TST) and, to a lesser extent, taste+value-based statements (T/V), alongside the most objective category (WOR). This pattern suggests that factual discussions occasionally incorporate both expressions of personal preference and claims about world states that approach fact-adjacent statements.

In contrast, Opinion-Discussion contexts (*AskPolitics*, *changemyview*, *prochoice*) show over-representation of value-based statements (VAL) and word+values statements (V/W), that is, opinion types that blend subjective judgment with appeals to shared values or factual claims.

Taken together with the patterns of Functional Dependence Relations observed in each Global Communicative Context in the previous Section, these results indicate that opinion statements grounded in intersubjective values are more likely to be explicitly discussed, eliciting direct expressions of agreement and disagreement.

However, given the relatively low frequency of the most subjective statements and the specific nature of the conversations in our corpus, these findings should be interpreted cautiously. They are not fully generalizable but can serve as preliminary indicators and guideposts for future research on subjectivity and interaction in online discourse.

7. Conclusions

We presented a corpus of Reddit conversations annotated for speech acts and Functional Dependence Relations. At the first level of annotation, we distinguished between fact and opinion statements, taking into account the amount and type of available evidence, and further characterized opinion statements along a five-degree scale of subjectivity.

Our analysis revealed systematic differences across communicative contexts. Opinion-Discussion contexts (*AskPolitics*, *changemyview*, *prochoice*) are characterized by a high frequency of opinion statements based on inter-subjective values, coupled with prominent Functional Dependence Relations of agreement and disagreement. In contrast, Information-Exchange contexts (*askscience*, *DoItYourself*, *todayilearned*) feature opinionated statements of lower subjectivity, with Functional Dependence Relations such as answers, continuers, and requests for clarification. These patterns highlight how discourse structure and interaction strategies are shaped by the type of communicative context.

Importantly, we fine-tuned a transformer model on the subjectivity scale, obtaining promising results, which opens the possibility for automatic or semi-automatic annotation. Such methods would allow us to extend the corpus and derive more generalizable conclusions. Future work also aims to annotate Functional Dependence Relations between segments within the same comment, capturing internal contribution structures, and further characterizing dialogue dynamics across different contexts.

8. Ethical Considerations and Limitations

The collection and annotation of Reddit comments prioritize user privacy by anonymizing all usernames and excluding original text. Users did not explicitly provide consent, as the data consists of publicly available content. The corpus has limitations in terms of generalizability: it includes only 47 conversations from a small set of specific subreddits, so findings may not extend to other Reddit communities or online discussions more broadly. Additionally, the relatively small size of the dataset constrains the representativeness of patterns observed. Finally, our study is restricted to English data, which may obscure cross-cultural variation in the interpretation of subjectivity. In particular, the assignment of categories such as VALUES or even the characterization of WORLD-related knowledge may depend on cultural and contextual factors. As such, the proposed annotation scheme may not transfer directly to multilingual or multicultural settings without further validation[6].

[6]We thank a reviewer for highlighting this issue. We also thank all reviewers, whose recommendations have substantially improved the quality of this paper.

9. Availability

The full corpus, along with the trained model of Section 5, is available at https://github.com/gFreijedo/arcc2026-subjectivity . To comply with Reddit's Terms of Service and GDPR regulations, ARCC does not redistribute raw comment text or original user identifiers. Usernames have been replaced with non-reversible, dataset-internal speaker IDs. The textual content of each segment can be reconstructed via the Reddit API using the provided comment identifiers and character offsets. The repository includes code for this reconstruction process.

It should be noted that some comments may no longer be accessible through the Reddit API due to deletion by users or moderators, which may prevent full reconstruction in certain cases.

10. Bibliological references

Alexandra Y. Aikhenvald and Alexandra Y. Aikhenvald. 2006. *Evidentiality*. Oxford University Press, Oxford, New York.

Francesco Antici, Federico Ruggeri, Andrea Galassi, Katerina Korre, Arianna Muti, Alessandra Bardi, Alice Fedotova, and Alberto Barrón-Cedeño. 2024. A Corpus for Sentence-Level Subjectivity Detection on English News Articles. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 273–285, Torino, Italia. ELRA and ICCL.

Nicholas Asher, Julie Hunter, Mathieu Morey, Benamara Farah, and Stergos Afantenos. 2016. Discourse Structure and Dialogue Acts in Multiparty Dialogue: the STAC Corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2721–2727, Portorož, Slovenia. European Language Resources Association (ELRA).

Nicholas Asher and Alex Lascarides. 2003. *Logics of Conversation*. Cambridge University Press.

Farah Benamara, Baptiste Chardon, Yannick Mathieu, and Vladimir Popescu. 2011. Towards Context-Based Subjectivity Analysis. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 1180–1188, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

Harry Bunt, Jan Alexandersson, Jae-Woong Choe, Alex Chengyu Fang, Koiti Hasida, Volha Petukhova, Andrei Popescu-Belis, and David Traum. 2012. ISO 24617-2: A semantically-based standard for dialogue annotation. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 430–437, Istanbul, Turkey. European Language Resources Association (ELRA).

Harry Bunt, Volha Petukhova, Emer Gilmartin, Catherine Pelachaud, Alex Fang, Simon Keizer, and Laurent Prévot. 2020. The ISO Standard for Dialogue Act Annotation, Second Edition. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 549–558, Marseille, France. European Language Resources Association.

Manish Kumar Chandan and Shrabanti Mandal. 2025. A comprehensive survey on sentiment analysis: Framework, techniques, and applications. *Computer Science Review*, 58:100777.

Elizabeth Coppock. 2018. Outlook-Based Semantics. *Linguistics and Philosophy*, 41(2):125–164.

Donka F. Farkas and Kim B. Bruce. 2010. On Reacting to Assertions and Polar Questions. *Journal of Semantics*, 27(1):81–118.

Anastasia Giannakidou and Alda Mari. 2021a. A Linguistic Framework for Knowledge, Belief, and Veridicality Judgment. *KNOW: A Journal on the Formation of Knowledge*, 5(2):255–293.

Anastasia Giannakidou and Alda Mari. 2021b. *Truth and Veridicality in Grammar and Thought: Mood, Modality, and Propositional Attitudes*. University of Chicago Press, Chicago, IL.

Jonathan Ginzburg. 2012. *The Interactive Stance: Meaning for Conversation*. Oxford University Press UK.

H. P. Grice. 1975. Logic and conversation. In Peter Cole and Jerry L. Morgan, editors, *Syntax and semantics: Vol. 3: Speech acts*, pages 41–58. Academic Press, New York.

Hugo Hernault, Danushka Bollegala, and Mitsuru Ishizuka. 2010. A Sequential Model for Discourse Segmentation. In *Computational Linguistics and Intelligent Text Processing*, pages 315–326, Berlin, Heidelberg. Springer.

Shafiq Joty and Tasnim Mohiuddin. 2018. Modeling Speech Acts in Asynchronous Conversations: A Neural-CRF Approach. *Computational Linguistics*, 44(4):859–894.

Shafiq Rayhan Joty, Giuseppe Carenini, and Raymond T. Ng. 2013. Topic Segmentation and Labeling in Asynchronous Conversations. *Journal*

of Artificial Intelligence Research, 47:521–573. ArXiv:1402.0586 [cs].

Lauri Karttunen and Annie Zaenen. 2005. Veridicity. In *Annotating, Extracting and Reasoning about Time and Events*, volume 5151 of *Dagstuhl Seminar Proceedings (DagSemProc)*, pages 1–9, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

Christopher Kennedy. 2013. Two Sources of Subjectivity: Qualitative Assessment and Dimensional Uncertainty. *Inquiry*, 56.

Manfred Krifka. 2022. Adjacency Pairs in Common Ground Update: Assertions, Questions, Greetings, Offers, Commands. In *Proceedings of the 26th Workshop on the Semantics and Pragmatics of Dialogue - Full Papers*, Dublin, Ireland. SEMDIAL.

Peter Lasersohn. 2005. Context Dependence, Disagreement, and Predicates of Personal Taste*. *Linguistics and Philosophy*, 28(6):643–686. Company: Springer Distributor: Springer Institution: Springer Label: Springer Number: 6.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. ArXiv:1907.11692 [cs].

John MacFarlane. 2005. The Assessment Sensitivity of Knowledge Attributions. In Tamar Szabó Gendler and John Hawthorne, editors, *Oxford Studies in Epistemology*, pages 197–234. Oxford University Press.

Jan Nuyts. 2001. Subjectivity as an evidential dimension in epistemic modal expressions. *Journal of Pragmatics*, 33(3):383–400.

Paul Portner. 2009. *Modality*. Oxford Surveys in Semantics and Pragmatics. Oxford University Press, Oxford, New York.

Marie-Paule Péry-Woodley, Stergos D. Afantenos, Lydia-Mai Ho-Dac, and Nicholas Asher. 2011. Le corpus ANNODIS, un corpus enrichi d'annotations discursives [The ANNODIS corpus, a corpus enriched with discourse annotations]. *Traitement Automatique des Langues*, 52(3):71–101. Place: France.

Roser Saurí and James Pustejovsky. 2009. FactBank: A corpus annotated with event factuality. *Language Resources and Evaluation*, 43:227–268.

Elena Savinova and Fermin Moscoso Del Prado. 2023. Analyzing Subjectivity Using a Transformer-Based Regressor Trained on Naïve Speakers' Judgements. In *Proceedings of the 13th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis*, pages 305–314, Toronto, Canada. Association for Computational Linguistics.

John Searle. 1995. *The Construction of Social Reality*. Free Press.

John R. Searle. 1969. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, Cambridge.

Tamina Stephenson. 2007. Judge dependence, epistemic modals, and predicates of personal taste. *Linguistics and Philosophy*, 30(4):487–525. Company: Springer Distributor: Springer Institution: Springer Label: Springer Number: 4.

Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26(3):339–374. Place: Cambridge, MA.

Kjell Johan Sæbø. 2009. Judgment ascriptions. *Linguistics and Philosophy*, 32(4):327–352.

Carla Umbach. 2016. Evaluative propositions and subjective judgments. In Cécile Meier and Janneke Van Wijnbergen-Huitink, editors, *Subjective Meaning*, pages 127–168. De Gruyter.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating Expressions of Opinions and Emotions in Language. *Language Resources and Evaluation*, 39(2):165–210.

11. Language Resource References

Freijedo Aduna, Gonzalo and Giannakidou, Anastasia and Mari, Alda. 2026. *Annotated Reddit Conversation Corpus (ARCC) for Speech Acts, Functional Dependence Relations, and Subjectivity*. ISLRN 00000000.

Qi, Peng and Zhang, Yuhao and Zhang, Yuhui and Bolton, Jason and Manning, Christopher D. 2020. *Stanza: A Python NLP Library for Many Human Languages*. PID https://stanfordnlp.github.io/stanza/.

A. Appendix A: Examples of Speech Act Annotations

The following examples illustrate the speech act categories used in our annotation scheme. Each entry corresponds to a sentence or clause from the corpus labeled with one of the defined categories. In the case of Assessment and Question (Opinion), we underline lexical markers of subjectivity (e.g., epistemic modals, evaluative adverbs).

Assessment

(11) You are definitely right that our human thinking can be our enemy here.

(12) That's probably the best outcome an English king could have hoped for.

Assertion

(13) Light is both an electric and magnetic wave.

(14) The United States does not recognize or consider Taiwan to be part of China.

Question (Opinion)

(15) [*Straight:*] Would you say "both interpretations are valid" about other key biblical relationships or events?

(16) [*Biased:*] Wouldn't you rather have genuine public discourse shape foreign policy than the other way around?

Question (Fact)

(17) [*Straight:*] What triggers a shark's blood sense/scent?

(18) [*Biased:*] Doesn't this require very pure water?

Advice/Suggestion/Recommendation

(19) Buy an extender from a big box store to move everything out to where you need it.

(20) Here is an example of someone selling one of the AM antennas on Etsy [URL], just to give you an idea of what they might look like.

Expressive

(21) Haha okay yeah I get it now thanks.

(22) Apologies if my understanding is incorrect.

Other

(23) Painting Wood Posts

(24) FYI

B. Appendix B: Examples of Functional Dependence Relations Between Segments

The examples below illustrate functional dependence relations annotated in our corpus, distinguishing between those that occur *between different comments* (inter-comment) and those that occur *within the same comment* (intra-comment). Each turn is labeled using a speaker identifier (A, B, etc.) followed by a numerical index indicating the sequential order of the speech act within that speaker's contributions (e.g., **A.2** refers to the second speech act from Speaker A).

Inter-comment functional dependence relations. These involve rhetorical moves that span across distinct speakers or comment turns. The following examples illustrate common inter-comment functional dependence relations.

Agreement and Disagreement

(25) a. **A.1:** The yeast eat the priming sugar inside the bottle...
b. **B.1:** That's true for basic home brewing.
c. **B.2:** Larger and industrial operations instead carbonate the beer after the yeast has pretty much worked through all the sugars in the brew.

***B.1** expresses Agreement with **A.1**; **B.2** contrasts with the generalization in **A.1**, expressing Disagreement.*

Request Clarification and Answer

(26) a. **A.1:** The mechanism is the same as you described either way...
b. **B.1:** Is it doing the exact same thing in beer, or is it different?
c. **A.2:** Exactly the same, yes.

***B.1** Requests Clarification of **A.1**; **A.2** responds with a direct Answer to **B.1**.*

Continuation -EX-

(27) a. **A.1:** Buy an extender from a big box store to move everything...
b. **B.1:** Thank you!

***B.1** provides a minimal Continuation -EX- follow-up to **A.1** across comments.*

Intra-comment functional dependence relations. These involve functional dependence relations internal to a single comment, typically reflecting internal discourse structure such as explanation, elaboration, contrast, continuation, or causal inference.

EXPLANATION

(28) a. **A.1:** My cat goes absolute apeshit for Subway's multigrain...
b. **A.2:** Turns out it's because there's tons of catnip oil...

A.2 offers an EXPLANATION/ELABORATION for **A.1** within the same comment.

CONTRASTIVE

(29) a. **A.1:** (Older) Android smartphones use the headphone cable. . .
b. **A.2:** But they can only pick up FM signals.

A.2 introduces a limitation that contrasts with the implication of **A.1**, establishing a CONTRASTIVE relation.

RESULT

(30) a. **A.1:** Many of them only have as much power as the people give them...
b. **A.2:** Which is why education and protest will be so important...

***A.2** presents a consequence of the proposition in **A.1**, marking a RESULT relation.*

CONTINUATION

(31) a. **A.1:** I just need to clarify a little bit of what you're saying...
b. **A.2:** You say they mean nothing,

***A.2** continues and specifies the content introduced in **A.1**, forming a CONTINUATION relation within the same comment.*

Constraints on Linking Element Choice in German Nominal Compounding: A Large-Scale Corpus Study

Maksim Shmalts
Department of Linguistics
University of Tübingen
maksim.shmalts@uni-tuebingen.de

Abstract

The N+N compound class is the largest and the most productive class of compounds in German. A significant number of N+N compounds insert a so-called linking element from a large inventory. The linker choice is notoriously irregular; instead of rules, it is governed by a set of constraints that can only limit this choice based on morphological, phonological, sometimes semantic and lexical properties of the first constituent. While constraints on linking element choice in German nominal compounding are extensively researched and well-documented, no large-scale corpus study has ever been reported on the subject of their empirical application. The present work aims at filling in this gap by conducting an extensive corpus study on potential and actual applicability of these constraints. The study summarizes 64 constraints collected from the relevant literature and obtains applicability statistics for 39 of them over 280k+ German N+N compounds. The study both confirms most of the evidence from previous literature and suggests novel evidence on German nominal compounding. It additionally highlights the importance of structured linguistic data for large-scale empirical studies.

Keywords: compounding, linguistic constraints, corpus study, German morphology

1. Introduction

Compounding is a process of word formation broadly defined as concatenation of two constituents with or without adding a linking element (also called linker) in between: Buch '*book*' + Deckel '*cover*' → Buch**ø**deckel '*book cover*' but Buch '*book*' + Regal '*shelf*' → B**ü**ch**er**regal '*bookshelf*'. The first constituent becomes the modifier, and the second the head of the resulting composite. In German, compounding is highly productive: thus, Baroni et al. (2002) report that in the APA newswire corpus (28M+ tokens), 47% of word types are compounds. Biconstituent nominal (N+N) compounds in German form the largest class of compounds (Ortner and Müller-Bollhagen, 1991, 53; Scherer, 2012; Schlücker, 2023). A sizable number of such compounds inserts an explicit linking element between the constituents. While German features "an unusually high number of different linking elements" (Schäfer and Pankratz, 2018), only one of them is grammatically correct for a given compound[1]. What makes the phenomenon notoriously complicated is the fact that it is often arbitrary which exact linker from the extensive inventory is the correct one. This choice is non-deterministic and cannot be predicted confidently from any properties of the constituents alone. On the contrary, it is governed by a large set of (irregular) *constraints* that can only limit this choice based on morphological, phonological, sometimes semantic and even lexical properties of the first constituent (Eisenberg, 2013, 227; Kürschner; Libben et al., 2009; Schäfer and Pankratz, 2018); the second constituent is argued to have no effect on linker choice[2] (Kürschner; Ortner and Müller-Bollhagen, 1991, 56).

Constraints on nominal compounding[3] in German have been extensively researched and well-documented in the linguistics literature over the last decades. Their irregularity and diversity is well-known and generally agreed upon. And yet, despite the high productivity of nominal compounding in German and the high complexity of the corresponding constraint space, no *large-scale* corpus study has ever been reported on the subject of *empirical* constraint application.

Our study serves as an initial attempt to investigate the empirical application of nominal compounding constraints in German quantitatively. More specifically, we focus on their *coverage* and *regularity*. By the coverage of a constraint, we mean the share of the units that satisfy the linguistic conditions for this constraint to be *potentially applicable*. In our terminology, such units are also referred to as units *covered* by this constraint. The regularity of a constraint is then the share of the units covered by this constraint that actually conform to it. In line with the introduced terminology, we call a constraint *actually applicable* to the units that conform to it, or simply say it *applies* to them. For instance, a dummy constraint (0) formulated

[1] Except for a small number of so-called doubtful cases when two variants coexist in language, see Nübling and Szczepaniak (2013).

[2] See exceptions in section 2.2.

[3] To clarify: by 'nominal compounding constraints', we mean constraints on choice of the linking element in N+N compounds.

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 14–38

as "First constituents that are constituted by nouns of the A class tend to attach -*x*-." will be potentially applicable to dummy compounds $A\mathbf{x}B$ and $A\mathbf{y}C$, but not to $C\mathbf{x}A$, and will be actually applicable only to $A\mathbf{x}B$. The formal definition of coverage and regularity follows in section 4.2.

We compute the coverage and the regularity of the constraints programmatically over a modification of a compound dataset originating from (Schäfer and Pankratz, 2018). Access to *strictly structured* linguistic data becomes a prerequisite for automated computation: only well-defined structures over the dataset can enable a uniform and reliable programmatic access to the target linguistic properties as defined by constraint applicability conditions. CELEX2 (Baayen et al., 1995) becomes the natural choice for us: this lexical database stores various linguistic information about its entries in a table-like format perfectly suitable for automated processing.

The present study makes two contributions to the developments in German nominal compounding studies: 1) Constraints on linker choice in German N+N compounds are collected from a number of relevant scientific sources, summarized and classified. 2) A corpus study is conducted on the matter of empirical application of the collected constraints[4]. The combined effect of these two contributions results in a systematic constraint space. Each constraint in the constraint space comes with the information about the type of linguistic information it employs (morphological, phonological, etc.), and, whenever possible, the measures of its coverage and regularity over 280k+ N+N compounds. This development is relevant both to descriptive studies in theoretical linguistics on German morphology and as a knowledge base/graph for novel compound generation with LLMs. The latter may be useful for evaluating linguistic competencies and reasoning abilities of LLMs and may increase the quality of generated compound datasets.

The paper is structured as follows. Section 2 summarizes principles of German nominal compounding. Section 3 gives an overview of previous work concerning German nominal compounding (in particular, compounding constraints). Section 4 introduces the methodology and the course of our empirical study. Finally, the results of the study are discussed and summarized in sections 5 and 6.

2. On Nominal Compounding in German

The N+N compound class is the largest and the most productive class of compounds in German. We consider solely N+N compounds because compounding in German conforms to binary recursion (Eisenberg, 2013, 219; Schäfer, 2018, 226-227), and so any compound can be decomposed into a chain of binary structures. While German (and Germanic in general) has a few constraints that specifically target compounds with three and more constituents[5] (see Krott et al., 2004; Kürschner; Wegener, 2005), we exclude them from the present study due to their small number. Furthermore, N+N compounds is the class where the explicit linking elements occur most often (Schlücker, 2023).

2.1. Linking Elements

The inventory of linking elements in German includes zero linker -*ø*-, the following explicit linkers: -*s*-, -*es*-[6], -*(e)n*-, -*e*-, -*"e*-[7], -*(")er*-[8], -*"*-, -*(e)ns*-, and a number of idiosyncratic linkers for loanwords. The latter — as well as linkers causing deletion of final stem segment(s) — are disregarded in the current study due to their insignificant frequency.

The correct linker is often not inferrable with certainty from any properties of the first constituent. The arbitrariness of choice may be so high that some nouns can attach different linkers in different compounds, e.g., Land '*state*' + Kreis 'district' / Regierung 'government' / Spiel 'match' → Land*ø*kreis '*regional district*' / Land**es**regierung '*regional government*' / L**ä**nd**er**spiel '*international match*'. Augst (1975, 134) finds 9.3% of entries have records with two different linkers and almost 1% of entries with three or four different linkers in a corpus of roughly 4k compounds. Rather than predict the correct linking element for a given noun, one can only limit the group of acceptable linkers, guided by a set of (irregular) constraints.

2.2. Constraints on Linker Choice

Constraints on nominal compounding in German are defined almost exclusively over linguistic information of the first constituent. Ortner and Müller-Bollhagen (1991, 56) document the single case

[4] Excluding semantic and lexical constraints, see more in section 4.4.

[5] Thanks to one of the anonymous reviewers for bringing that to our attention.

[6] -*s*- and -*es*- are not notated as -*(e)s*- as they are not allomorphic (-*es*- is isolated, see Eisenberg, 2013, 229; Kürschner, 2010), in contrast to -*(e)n*- and -*(e)ns*-.

[7] *"* signals that the linker triggers umlauting of the stem vowel of the first constituent.

[8] To the best of our knowledge, the -(")er- linker triggers umlauting necessarily whenever the stem vowel may undergo it. We therefore do not distinguish between -er- and -"er- (as opposed to -e- vs. -"e-) as cases like Kind**er**jacke '*children's jacket*' may be accounted for by assuming that -(")er- tries to apply umlauting but the target vowel cannot not be umlauted. We hold the same assumption against the -(")er plural marker.

when the properties of the second constituent contribute equally to linker choice, see constraint (39) in the list of example constraints below. Apart from that, Eisenberg (2013, 227) suggests that sometimes, the semantic relation between the two constituents may play a definitive role in linker choice. This claim is supported by evidence provided by Koliopoulou (2014), also Neef and Borgwaldt (2012), who independently formulate constraint (36).

The rest of the constraints refer solely to the properties of the first constituents and are based on various types of linguistic information. The literature documents morphological, derivational[9], phonological, semantic, lexical, and mixed constraints. Below is an example list of constraints collected from Eisenberg (2013, 227-230) and Ortner and Müller-Bollhagen (1991, 56-57, 83, 89-94, 109).

(7) [morphological] First constituents that are constituted by nouns that build the plural form with *-(")er* mostly attach *-ø-* or *-(")er-*: Buch '*book*' + Deckel '*cover*' → Buch**ø**deckel '*book cover*' but Kind '*child*' + Alter '*age*' → Kind**es**alter '*childhood*'

(22) [derivational] First constituents that are constituted by deadjectival feminine nouns with a schwa suffix almost always attach -(e)n- or a zero linker; each of the two linkers is preferred in about the same number of cases: Hitze '*heat*' + Welle '*wave*' → Hitze**ø**welle '*heatwave*' but Liebe '*love*' + Brief '*letter*' → Liebe**s**brief '*love letter*'

(36) [semantic] Copulative compounds insert *-ø-* regularly (by copulative compounds are understood compounds in which the two constituents do not exhibit a clear modifier-head relation): Dichter '*poet*' + Komponist '*composer*' → Dichter**ø**komponist '*poet-composer*'

(39) [semantic, lexical] Compounds whose first constituent designates a person and whose second constituent is one of 'Mann', 'Frau', 'Leute', 'Tochter', 'Gattin', or 'Witwe' tend to insert *-s-*: Reiter '*rider*' + Mann '*man*' → Reiter**ø**mann '*horseman*'

(46) [morphological, phonological] Almost all feminine nouns that constitute first constituents that attach -s- are polysyllabic: Arbeit '*work*' + Tag '*day*' → Arbeit**s**tag '*working day*'

As will be demonstrated in sections 4.1 and 5, nominal compounding constraints in German are highly diverse in terms of their regularity and coverage, as well as regarding the type of linguistic properties they target.

[9]To the best of our knowledge, this term is not well-established. By derivational constraints we mean those that target classes of nouns with specific derivational structure, e.g., nouns with specific affixes, deverbal nouns, etc.

3. Previous Research

Linking elements in (N+N) compounds have been a subject of interest in German morphology for decades. Numerous works address different aspects of their development, functionality, and linguistic status.

Nübling and Szczepaniak (2013), Wegener (2003), Wegener (2005) investigate the origin and the diachronic development of the linking elements. They identify two sources of German linking elements: Germanic primary suffixes and later genitive markers. By addressing the process of their lexicalization and reanalysis, the authors explain the restrictions on linker choice in contemporary German from a historical perspective. In particular, they suggest that plural markers *-(e)n*, also *-e*, *-"e*, *-(")er* originate from the primary suffixes, and the homonymous linking elements evolve either from these plural markers (Nübling and Szczepaniak, 2013), or in parallel to them (Wegener, 2003). This way, these works motivate the fact that these linkers appear exclusively with nouns of the corresponding plural[10].

A larger share of the literature body reports contemporary limitations on linker choice without focus on their origin. For example, Eisenberg (2013, chapter 6.2), Ortner and Müller-Bollhagen (1991, chapter A.5), Schäfer (2018, chapter 8.1) expound general principles of German (nominal) compounding, each covering acceptability conditions of most of the investigated linkers. Numerous works are dedicated specifically to the *-s-* linker since it is the only productive linker not bound to any inflectional class: Libben et al. (2009), Kopf (2017), Kürschner, Nübling and Szczepaniak (2008).

Finally, a notable branch of research is oriented towards investigating various analogical effects associated with compounding in German. Thus, Krott et al. (2007) find that choice between *-s-*, *-(e)n-*, and *-ø-* is biased towards the most frequent linker occurring with the given first constituent in other compounds, with a statistically significant effect. With different experimentation course, Neef and Borgwaldt (2012) arrive at similar results and confirm the left constituent bias. They also suggest that building novel compounds is subject to inter-speaker variability to a certain degree. On another note, Schäfer and Pankratz (2018) provide evidence that linkers identical to the plural markers may serve a clue to a plural or collective meaning of the first constituent in the compound.

[10]For -(e)n- and -e-, Nübling and Szczepaniak (2013) also identify edge cases not related to the plural suffixes.

All these findings are of undeniable value and importance. Combined, they provide an exhaustive yet granular overview of the constraints on linker choice in German. However, the absolute majority of these works have a significant fallback: while noting the infamous irregularity of these constraints, they fail to deliver its empirical measures. This results in quite vague constraint definitions that sometimes hinder building an adequate idea of a certain constraint. A few examples follow[11]: "If the stem [of a feminine noun] is monosyllabic, the *-en-* linker is *sometimes* inserted, *sometimes* not." (Eisenberg, 2013, 229); "There is an *observable tendency* to insert *-e-* after nouns designating animals [...] with many counterexamples." (Nübling and Szczepaniak, 2008, 11); "Strong and mixed masculine and neuter nouns *often but by no means always* insert *-s-* or *-es-*[12]" (Schäfer, 2018, 229).

Clearing up such vague definitions would only be possible by the means of a dedicated corpus study. However, only targeted corpus studies covering single constraints have been made so far. Kopf (2017), Kürschner, and Nübling and Szczepaniak (2008) aim at investigating the increased probability of *-s-* after prefixed deverbal nouns. They confirm that prefixed deverbals attach *-s-* in a majority of cases: 67.5%, 76.5-78%, and 85% (with unstressed prefix) / 36% (with stressed prefix), respectively. Going further, Nübling and Szczepaniak (2008) and Wegener (2005) address the relation between the probability of *-s-* and the sonority of the last segment of the noun. They find that *-s-* occurs after plosives in 15-20% and after sonorants in 1.8-4.7% of cases, and it never occurs after a full vowel.

To the best of our knowledge, no attempts have been reported yet of extensive corpus studies that would cover the entire space of nominal compounding constraint or at least a large part of it. We therefore aim at filling in this gap by conducting a novel large-scale corpus study on the application of nominal compounding constraints in German.

4. Empirical Study

The present study was conducted in four steps. We briefly summarize them here, and introduce them in more detail in the corresponding subsections 4.1-4.4 below.

Step one fixes the definitions and linguistic status of constraints from the relevant literature.

Step two ensures reproducibility of the study. It establishes formalizations concerning potential/actual applicability of the collected constraints and calculation of their coverage and regularity.

The third step prepares the data for the corpus experiments. It obtains a massive collection of N+N compounds taking a compound dataset DeCOW16AX-comps originating from Schäfer and Pankratz (2018) as a baseline. It then derives a custom noun database from CELEX2 (Baayen et al., 1995) — a large lexical database for English, German, and Dutch that stores various linguistic properties of its entries (lemmata and wordforms). Additionally, it retrieves word counts of the entries of the two obtained modifications from DeReKo (IDS, 2026) — the largest collection of text corpora in contemporary German.

Step four conducts the corpus study itself. It estimates the empirical applicability of the collected constraints over our modification of DeCOW16AX-comps following the procedural formalization from the second step.

The developments of the empirical study — including the collected constraint definitions and supporting information, the compound dataset, and the relevant codebase — are published under `https://github.com/maxschmaltz/DieKonstraints/tree/comp_constr_corp_study_mar26`. Please refer to the repository's README to navigate the project files.

4.1. Step One: Constraint Acquisition

To obtain linker choice constraints reported in the linguistics literature so far, we studied a number of scientific works dedicated to German (nominal) compounding. We aimed at keeping the collection of scientific sources diverse in terms of their format, purpose, object of research, and date of publication. In case of discrepancies, we did not take sides with one or another analyses and kept both sources, assuring a wide spectrum of linguistic perspectives. The resulting body of literature spans over 30+ years and includes chapters of fundamental books on German morphology in general and on (nominal) compounding in particular, survey-like papers reporting generalizations of German compounding and linker choice, as well as more targeted papers addressing specific aspects of compounding in German, such as their functionality and grammatical status, restrictions on their insertion, etc. To adopt a broader perspective, we also involved works dedicated to the origin of linking elements, analogical effects in compounding, and comparative studies on linking elements in Germanic. The resulting set of works includes 18 sources and is provided in Appendix A.

From these sources, we extracted claims that described restrictions on, or patterns of, choice of the linking elements in N+N compounds. We find no critical contradictions across the works. The

[11] Whenever necessary, translated by the author of the present study. In italics, we highlight the vague spots.

[12] The original work reads *-(e)s-* but as mentioned before, we distinguish between these two linkers.

only disagreements concern regularities or linguistic nature of a few constraints. These are of no significance to us, since we obtain our own statistics of constraint application. Otherwise, there is an ongoing debate on morphological (Kürschner) vs. phonological (Nübling and Szczepaniak, 2008) conditioning of *-s-*. We simply formulate both options as two separate constraints for further verification.

In total, we summarized 64 constraints. Apart from the default constraint striving to assign a zero linking element to any given compound, two main categories of constraints could be established. The first category is prevalent with 41 constraints. It limits its acceptable linkers for a given noun class ('*A* can attach either *x* or *y*'), as exemplified by constraints (7), (22), (36), (39) above; we therefore call it *P2L* (properties to linkers). The second category (22 constraints) goes the opposite way: it puts restrictions on the noun classes that can attach a given linker ('*x* can only be attached by *A* or *B*'), as exemplified by constraint (46); correspondingly, we name it *L2P*. Table 2 exhibits constraint types by level(s) of accessed linguistic information[13]. Both categories actively employ all types of linguistic information, involving prevalently morphological but to a noticeable extent derivational, phonological, and semantic properties. That observation is in line with the fact that linker choice is highly variable and is affected by a large variety of factors.

Fifty-nine out of 64 constraints are defined over the properties of the first constituent. Note that the 22 L2P constraints which all belong to that group additionally require insertion of a certain linker to be potentially applicable. For instance, constraint (46) covers not any feminine first constituent but only those that attach *-s-*. From the remaining five constraints, one depends on the properties of both constituents, three on the semantic relation between them, and the last one is the default constraint that is potentially applicable to any given constituent combination. This finding confirms the general agreement that the linker choice depends almost exclusively on the properties of the first constituent. In the following, the component(s) of a compound accessed by a constraint will be called the target *item* of this constraint in the compound. For most constraints, the first constituent of a compound or the combination of the first constituent and the linker is the target item. For the five remaining constraints mentioned above, the combination of both constituents is the target item.

The full list of constraint definitions, properties, and supporting information (P2L/L2P type, employed linguistic information, target item, examples, and counterexamples) is available in Appendix E.

Type	P2L	L2P
morphological	6	6
+ derivational	1	1
+ phonological	1	0
+ phonological	2	1
+ semantic	3	1
+ lexical	0	1
+ semantic	8	1
+ lexical	0	1
derivational	9	2
+ semantic	2	0
phonological	5	4
semantic	2	0
+ lexical	1	0
lexical	1	4

Table 1: Types of nominal compounding constraints by employed linguistic information. Mixed types are marked by indentation. Mid-grey marks types of zero, and light-grey of up to three constraints.

4.2. Step Two: Procedural & Mathematical Formalizations

Every constraint definition consists of two components. The first component declares the linguistic properties of the constraint's target item that determine potential applicability of this constraint to an arbitrary compound formed from it. For most constraints, the first component is effectively the description of the desired properties of the first constituent. The second component specifies a set of acceptable linkers that are expected to appear in a compound built with this item if the constraint applies. For instance, constraint (7) (target item: first constituent) may be represented[14] as `[N1-pl = (")er] → [-(")er-, -∅-]`. Constraint (39) (target item: the combination of both constituents) can similarly be formulated as `[N1 is pers][N2 ∈ {Mann, Frau, ...}] → [-∅-]`. Any constraint can then be abstracted as `[cond 1]...[cond` n`] →` `[link 1, ..., link` m`]`. Then, the formal tests for potential and actual applicability are trivial. For an arbitrary constraint (X) and an arbitrary N+N compound C formed from item I, it holds: 1) (X) is potentially applicable to C if for $i = (1, ..., n)$, **all** `[cond` i`]` defined over I are true; 2) (X) applies to C if (X) is potentially applicable to C and

[13] Derivational constraints defined over morphologically restricted classes are not additionally marked as morphological if they target nouns with suffixes that determine morphological properties of the derived noun (e.g., suffix *-heit* always produces weak feminines).

[14] The following abbreviations are adopted for these examples: `N1` and `N2` stand for the first and the second constituent, respectively, `pl` stands for 'plural marker', `pers` — for 'person designation', `cond` — for 'condition', `link` — for 'linker'.

if for $j = (1, ..., m)$, C inserts **any** of `[link j]`[15]. Please note that since all `[cond i]` are defined over the properties of I, potential applicability of (X) to C and to further compounds built from I will be identical. This is not the case with actual applicability which is determined by the linker in a specific compound. For the L2P constraints, the two constraint definition components and, correspondingly, the tests are swapped.

When calculating the statistics of the empirical constraint application, we will distinguish between *type coverage and regularity* and *item coverage and regularity*[16]. This distinction results from calculating the application statistics over unique compounds in the dataset or over unique items (mostly first constituents) that form these compounds, respectively. A characteristic example may be helpful to motivate this distinction. As discovered after conducting the corpus study, constraint (22) (target item: first constituent) is potentially applicable to 2798 compounds that are built from 32 unique first constituents. All 414 of 2798 compounds that violate (22) are built with one single first constituent Liebe '*love*' (Liebe**s**brief '*love letter*', Liebe**s**erklärung '*declaration of love*', etc.). That means that while $\sim 97\%$ unique items (31 out of 32 covered first constituents) always build compounds that conform to (22), they only account for $\sim 85\%$ of unique compounds. In other words, (22) is (almost) regular from the item perspective and is only a tendency from the compound perspective. As will be elaborated in section 5, there are further constraints that exhibit a similar gap. We therefore report two variants of coverage and regularity: the *type* variant and the *item* variant[17].

Formally, *type coverage* of a constraint (X) amounts to the ratio between the number of unique compounds to which (X) is potentially applicable as defined in the corresponding formal test, and the total number of unique compounds in the dataset. Since potential applicability of (X) to compounds built with the same item is identical, *item coverage* of (X) is defined simply. It equals the ratio between the number of unique items that build the compounds covered by X, and the total number of unique items found in the dataset.

Type regularity of (X) amounts to the ratio between the number of unique compounds to which (X) applies as defined by the respective formal test, and the number of unique compounds covered by (X). As opposed to potential applicability, (X) may (and often does) apply only selectively to different compounds built with the same item. To address this peculiarity, item regularity examines groups of compounds built with every unique item separately. Within each group, *group regularity* is calculated as the ratio between the number of unique compounds in the group that conform to (X) and the total number of unique compounds in the group. *Item regularity* of (X) is then the average over all group ratios. Item regularity δ of (X) thus conceptually means the following: on average, for any arbitrary item that satisfies linguistic conditions put by (X), $\delta \times 100\%$ of all compounds formed from this item conform to (X), regardless of the productivity of the item. Consequently, this method eliminates potential item productivity bias against type regularity exemplified by the first constituent Liebe with respect to constraint (22).

To ultimately clarify the provided mathematical formalizations, we demonstrate the calculation of the applicability statistics for constraint (22) in Appendix B. In the following, the item variant of coverage and regularity will be considered principal and will be reported first, followed by the type variant. This is motivated by the fact that, as demonstrated above, the item variant is not affected by the item productivity bias and is therefore more balanced and illustrative. Finally, we adopt the term of *combined coverage and regularity* for the contexts in which we simultaneously compare both types of the measures to some absolute value. Combined coverage/regularity $> \gamma$ of (X) means that both item and type coverage/regularity of (X) exceed γ. Correspondingly, combined coverage/regularity $< \gamma$ points that both types of coverage/regularity lie below γ.

4.3. Step Three: Data Acquisition

As pointed out in section 4, we employed three data sources for various purposes. DeCOW16AX-comps[18] was used to derive a massive N+N collection that served the corpus over which we assessed the empirical applicability of the constraints. CELEX2[19] was accessed during the experimental

[15] Note that if a constraint implicitly licenses further linkers, these are not accepted in our procedure, as these are exactly the irregular deviations that we want to record. For example, even though constraint (39) only *tends* to insert *-s-* — and so allows other linkers —, we only consider that it applies if attached is the *-s-* linker.

[16] Please note that this distinction is not the same as the customary distinction between type and token.

[17] In a dedicated branch of the study, we additionally calculated token coverage and regularity of the constraints that based on word counts of compounds. However, the values of the type and the token variants diverged insignificantly, and so the token variant was excluded from the present study.

[18] DeCOW16AX-comps is licensed under CC BY-NC 4.0 and are freely available for research purposes.

[19] CELEX2 is distributed under a custom license authorizing unrestricted usage for research purposes, see https://catalog.ldc.upenn.edu/docs/LDC96L14/celex.readme.html#Copyright. The data was obtained in compliance with this license.

phase to test for various linguistic information of compound constituents. Finally, DeReKo [20] supplied the word counts for the entries of our subsets of the first two. The preprocessing procedure of DeCOW16AX-comps depended on that of CELEX2, so we start the overview over data acquisition with the latter.

CELEX2 contains ∼51k German lemmata. For each lemma, it provides a rich set of linguistic information. Relevant for the present study are the records about POS, gender, inflectional paradigm, morphemic structure (both base word and affixes), phonetic transcription, and syllabification scheme of the lemmata. The high quality of annotations in CELEX2 minimized the number of preprocessing steps undertaken. Apart from a few minor preprocessing steps (mostly concerning data formatting), we simply extracted a subset of one-stem nouns that met two requirements: 1) they had the full set of the relevant records enlisted above; 2) their word count in DeReKo reached the minimum threshold of 50 occurrences. For word counts, we were querying a collection of DeReKo subcorpora compiled under the `copr-d` (Korpora aus Deutschland '*corpora from Germany*') namespace of the KorAP interface of DeReKo with a total of 15B+ tokens. This, together with the minimal creation year set to 1970, allowed us to avoid overt regionalisms or anachronisms. The frequencies were obtained via the KorAP OpenAPI endpoint (`https://korap.ids-mannheim.de/api/v1.0/openapi/`), the endpoint was accessed in early February 2026. The resulting sample of CELEX2 (henceforth also called CELEX-nouns) contains ∼12k nouns.

DeCOW16AX-comps supplied us with ∼22.3M SMOR (Schmid et al., 2004) analyses of German compounds with nominal heads. We took this collection as a starting point and in a few steps, dropped all records unsuitable for our study. The resulting modification that we called GeCoDB_v06 contains ∼280k N+N compounds featuring all variety of the linkers considered in the present study. Each record is a triplet identifying the two constituents and the linker of the compound. We only kept compounds where both constituents are present in CELEX-nouns — that ensured that we had access to the full set of required linguistic information for all of our compounds. To further ensure a high quality of the sample, we set a minimum word count threshold of 20 in DeReKo[21], and for its modifier to form at least ten compounds in GeCoDB_v06 with different heads.

[20]Most of the DeReKo subcorpora are licensed under a QAO-NC license. DeReKo may be used freely for scientific purposes.

[21]Word counts were collected under the same procedure that was utilized for CELEX2.

4.4. Step Four: Corpus Experiment

To be able to estimate constraint applicability over the GeCoDB_v06 corpus, we translated the procedural formalizations of the tests for potential and actual applicability from section 4.2 into Python code. The checker for potential applicability of a certain constraint accepts a compound and accesses the CELEX-nouns database to test for every condition `[cond` i`]` enclosed in its formalization. The checker for actual applicability simply compares whether the linker indicated in the input compound by SMOR is one of the licensed `[link 1, ..., link` m`]`. For the L2P constraints, the programmatic checkers are swapped as well, following the formal tests. Since we could only access morphological, derivational, and phonological properties of single constituents in CELEX-nouns, constraints involving semantic or lexical information of the first/second constituent as well as the semantic relation between the two were excluded from automated processing. In total, pairs of checkers for 40 constraints were implemented.

We ran all available checkers against every compound in GeCoDB_v06. As a result, we automatically discovered and marked all compounds to which the corresponding constraints are potentially and actually applicable (individually). Finally, we employed the mathematical formalizations from section 4.2 to calculate the two variants of coverage and regularity for each constraint.

We find it crucial to emphasize the role of the structured linguistic data for our study. The strict triplet structure of SMOR analyses in GeCoDB_v06 allowed to establish a dense connection to the CELEX-nouns database through reliable lookup of compound constituents in CELEX-nouns. In its turn, the highly structured tabular format of CELEX-nouns played a principal role in stable and predictable programmatic access to the desired linguistic properties of the constituents. It is therefore the structural concert of the two data collections that were crucial for reliable automated testing for empirical constraint application in the course of our corpus study.

5. Results & Discussion

In total, we obtained item and type coverage and regularity for 39 constraints[22] (23 P2L, 15 L2P, and the default constraint). Figure 1 demonstrates the distribution of the obtained values. The entire constraint space with the mapping is depicted illustratively in Appendix D, and the precise item and type

[22]One of the 40 constraints selected for automated processing did not cover any compounds in the dataset (namely, (26), see Appendix E).

coverage and regularity values for each analyzed constraint are provided in Appendix E.

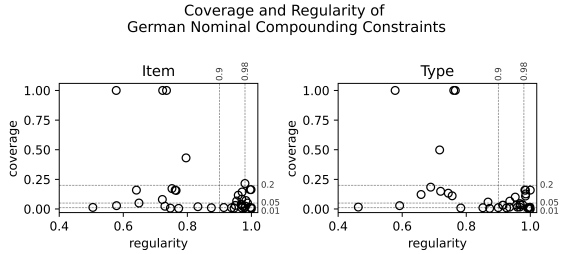


Figure 1: Distribution of the obtained coverage and regularity values. Each constraint appears twice: on the left and on the right.

Both coverage and regularity values fluctuate significantly. Item and type coverage stretch over the entire range of values: they effectively approximate 0.0 for a few constraints and reach 1.0 for three constraints. The standard deviation (SD) of 0.26 and 0.27 for the two variants, respectively, further confirms a large spread of values. Nevertheless, it can be observed that most of the coverage values concentrate at the bottom part of the plot. Indeed, 34 out of 39 constraints lay below the threshold of 0.2 for both item and type coverage. Going further, combined coverage falls under 0.05 for almost half of the inspected constraints (19 in total). Three constraints cover very restricted groups of nouns with combined coverage amounting to < 0.01.

Regularity, on the contrary, is distributed more compactly. All item and type regularity values lie in a range $(0.46; 1.0]$. They are also spread more evenly than the coverage values, with significantly lower SD of 0.14 for both variants. Even though the full range of values is densely occupied with no significant gaps, there is still a visible shift of values towards the right edge of the axis. Thus, slightly over half of the analyzed constraints (20 in total) reach a combined regularity of over 0.9. However, only six of them appear to be approximately regular with a combined regularity of > 0.98.

We observe that moderate and low values of coverage and regularity of both types are distributed relatively evenly between the P2L and the L2P constraints, whereas high values are distributed in a nearly complementary manner. Specifically, the entire range $(0.02; 1.0]$ of combined coverage values is occupied by three P2L constraints plus the default constraint. On the other hand, the upper range $(0.98; 1.0]$ of combined regularity is populated almost exclusively by the values of L2P constraints, with five out of six constraints in this range belonging to this category. These observations suggest that the restrictions on the noun classes that may attach certain linkers gravitate towards reliable application, while the patterns of linker choice prioritize broader predictive scope. The plot of applicability statistics of the P2L vs. the L2P constraints is provided in Appendix C.

A productivity bias as exemplified by constraint (22) is detected for six constraints in total. Type regularity of constraints (4), (22), and (42) is at least 10% lower than their item productivity. Similarly to (22), the bias in (4) and (42) is produced by smaller sets of items that productively form compounds that violate these constraints. Type regularity of constraints (19), (25), and (44) is, on the contrary, higher than their item regularity by at least 10%. This points to an opposite situation: the bias comes from sizable sets of items that produce small groups of compounds that tend to violate these constraints. For instance, 140 out of 1517 covered compounds that violate (19) are formed from 8 out of 27, or almost a third of unique first constituents.

No correlation between coverage and regularity is observed, neither for the item nor for the type variant. We conclude that these two properties are independent.

To assess the extent to which the collected constraints are supported by the empirical data, we assigned a threshold of minimal item regularity for each unique quantifier found in the constraints definitions ("mostly", "regularly", etc.). Item regularity was chosen as a more balanced metric as compared to type regularity. As a result, the scale demonstrated in figure 2 was established[23]. A constraint was considered *supported* if its item regularity equaled or exceeded the threshold corresponding to its textual quantifier (e.g., "usually" vs. item regularity of ≥ 0.9). A constraint was considered *partially supported* if its item regularity equaled or exceeded the threshold of the previous level on the scale but not its own (e.g., "usually" vs. item regularity in range $[0.6; 0.9)$). A constraint was regarded as *not supported* if its item regularity was below the threshold of the previous level (e.g., "usually" vs. item regularity of < 0.6).

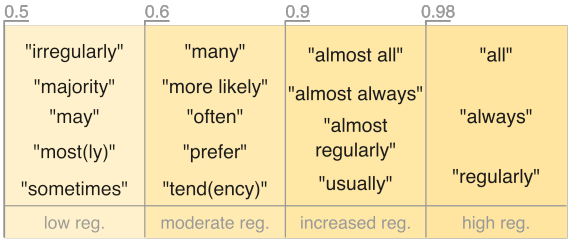


Figure 2: Quantifiers and the corresponding item regularity thresholds by regularity (reg.) levels.

Most of our findings corroborate the qualitative evidence from previous literature. Specifically, 28 out of 39 analyzed constraints are supported by the

[23] Please note that the threshold for the regular level is reduced to 0.97 for constraint whose item coverage equals or falls under 0.01. The reduction counterbalances an increased sensitivity of item regularity to counterexamples resulting from a very small number of items.

corpus data as defined in the previous paragraph. In the next paragraphs, we inspect the constraints that are partially/not supported by the empiric data on a number of illustrative examples.

Seven constraints marked as highly regular in the literature and one constraint marked as moderately regular did not confirm their status on massive data but appeared to be partially supported. These are P2L constraints (3), (29), (31) and L2P constraints (47), (48), (52), (56), (61). For instance, constraint (3) which is formulated as "First constituents that are constituted by nouns that end in a full vowel always adopt *-ø-*" (Kürschner, 110; Schäfer and Pankratz, 2018, 9) is consistently violated by compounds built with foreign nouns ending with a stressed *-ee*/*-ie* (e.g., Idee**n**buch '*book of ideas*', Kalorie**n**gehalt '*calorie content*', Melodie**n**folge '*melody sequence*'). Otherwise, rare counterexamples with native words are found, such as Schuh**e**kauf '*purchase of shoes*'. The other seven constraints from this group likewise have numerous counterexamples.

Constraints (19), (25), (33) (all P2L) were not supported by the empirical data. The first disproved constraint (19) is formulated in Ortner and Müller-Bollhagen (1991, 83, 107): "First constituents that are constituted by derived nouns with suffixes *-bold*, *-nis*, *-rich*, *-at*, *-al* that build the plural form with *-e* attach *-ø-* regularly." With over 190 counterexamples of compounds with first constituents ending with *-at* and attaching *-s-* (e.g., Dekanat**s**veranstaltung '*deanery event*', Referat**s**vorbereitung 'presentation preparation'), we record item regularity of 0.747 and type regularity 0.873 for this constraint. For the covered compounds with the rest of the suffixes (excluding *-at*), the constraint is regular, indeed. The second not supported constraint (25) is formulated as "First constituents that are constituted by deverbal nouns that end in suffix *-en* attach *-s-* regularly." (Eisenberg, 2013, 230; Nübling and Szczepaniak, 2013, 78). The counterexamples include two groups of compounds. The first group consists of 104 *ø*-linked variants of existing compounds conforming to (25). In most of these pairs, the *ø*-linked variant has a significantly lower frequency (e.g., Leben**s**stil vs. Leben*ø*stil '*lifestyle*' with DeReKo word counts 63234 and 293, respectively). This observation might signalize that constraint (25) may be evolving towards lower restrictiveness in contemporary German. The second group includes 82 further compounds with *-ø-* formed from 13 first constituents (Beben*ø*gebiet '*quake zone*', Braten*ø*soße '*gravy*', Husten*ø*saft '*cough syrup*', etc.). Overall, the constraint reaches only 0.773 of item regularity (but 0.926 of type regularity), which corresponds to the moderate regularity level. Finally, constraint (33) "First constituents that are constituted by feminine nouns that end in schwa adopt *-(e)n-* regularly." (Eisenberg, 2013, 229; Krott et al., 2007, 3) is not supported by the data. Even though Fuhrhop and Kürschner (2015, 573), Kürschner, 112, Ortner and Müller-Bollhagen (1991, 93) point out that *-ø-* is also probable after such nouns if the schwa is a deadjectival or a deverbal suffix, it seems that the real number of such cases in language has been underestimated in the literature. We detected over 5.3k compounds with 109 various deverbal and deadjectival feminine first constituents that do not insert an explicit link (e.g., Reise*ø*ziel '*destination*', Glätte*ø*stelle '*slippery spot*'). Even apart from the derived first constituents, a very large group of counterexamples formed from 203 non-derived first constituents (Analyse*ø*bericht '*analysis report*', Bronze*ø*lack '*bronze lacquer*', Ehe*ø*mann '*husband*', Messe*ø*besucher '*visitor of a fair*', and much more) adds to this significant number, resulting in over 9.6k compounds violating the constraint. In total, this constraint achieves an item regularity of only 0.765 and a type regularity of 0.744, which, again, belongs to the moderate regularity level.

The provided examples reveal that for some constraints, not the regularity level but the conditions on potential applicability seem to be formulated incorrectly. Consider constraint (19). As already mentioned, it applies regularly to compounds with first constituents ending in all suffixes specified in its definition, besides *-at*. Correspondingly, after removing this suffix from the definition, (19) would be fully supported by the empirical data. In Appendix F, we provide a list of our suggestions on how to correct the definitions of the constraints that are supported only partially or not supported. We additionally implemented dedicated Python checkers for the corrected versions of constraint as proposed by us, and we report that all corrected versions are supported by the corpus data.

6. Conclusion

The linker choice in German N+N compounds is highly unpredictable. It is governed by a set of (ir-regular) constraints whose conditions on potential applicability are defined over various linguistic domains. The present study collects constraints documented in the linguistics literature and empirically investigates their application over a massive compounds corpus. The study verified 39 constraints collected in the relevant literature. It confirmed 28 constraints, found smaller inconsistencies towards the corpus data in eight constraint definitions, and identified major discrepancies towards the data for three constraints. In sum, it proposes novel findings on German nominal compounding and calls attention to the necessity to support qualitative linguistic studies by empirical estimations.

7. Limitations

In future research, we plan to implement the missing semantic and lexical constraints and obtain a full picture of nominal constraints' coverage and regularity. An open question is whether the type of linguistic information that a constraint employs has any relation to the constraint's coverage and regularity. Finally, during the experiment we detected a large number of constraint conflicts over various compounds. We believe that investigating constraint conflict resolution is a crucial research direction for understanding the complex interaction of the constraints in the process of linker choice. This research question is therefore to be addressed in future studies.

8. Acknowledgments

I would like to thank my PhD advisor, Erhard Hinrichs, for extensive comments on the paper, insightful discussions, and guidance throughout the study. His academic support was crucial to the success of this project. I am grateful to the three anonymous reviewers for the helpful comments. I would like to acknowledge the contribution of Marie Hinrichs, who did the final proofreading. Any remaining errors are solely my responsibility.

9. Bibliographical References

Gerhard Augst. 1975. *Untersuchungen zum Morpheminventar der deutschen Gegenwartssprache*. Narr, Tübingen.

Marco Baroni, Johannes Matiasek, and Harald Trost. 2002. Wordform- and Class-based Prediction of the Components of German Nominal Compounds in an AAC System. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Peter Eisenberg. 2013. *Grundriss der deutschen Grammatik*, 4 edition. J.B. Metzler Stuttgart, Stuttgart. EBook published 27 August 2016.

Nanna Fuhrhop and Sebastian Kürschner. 2015. *32. Linking elements in Germanic*, pages 568–582. De Gruyter Mouton, Berlin, München, Boston.

Maria Koliopoulou. 2014. How close to syntax are compounds? Evidence from the linking element in German and Modern Greek compounds. *Rivista di Linguistica*, 26(2):51–70.

Kristin Kopf. 2017. *Fugenelement und Bindestrich in der Compositions-Fuge*, pages 177–204. De Gruyter, Berlin, Boston.

Andrea Krott, Gary Libben, Gonia Jarema, Wolfgang Dressler, Robert Schreuder, and Harald Baayen. 2004. Probability in the Grammar of German and Dutch: Interfixation in Triconstituent Compounds. *Language and Speech*, 47(1):83–106. PMID: 15298331.

Andrea Krott, Robert Schreuder, R. Harald Baayen, and Wolfgang U. Dressler. 2007. Analogical effects on linking elements in German compound words. *Language and Cognitive Processes*, 22(1):25–57.

Sebastian Kürschner. Verfugung-s-nutzung kontrastiv. Zur Funktion der Fugenelemente im Deutschen und Dänischen. *Tijdschrift voor Skandinavistiek*, 26(2).

Sebastian Kürschner. 2010. Fuge-n-kitt, voeg-en-mes, fuge-masse und fog-e-ord - Fugenelemente im Deutschen, Niederländischen, Schwedischen und Dänischen : ein Grenzfall der Morphologie im Sprachkontrast. (Linking elements in German, Dutch, Swedish, and Danish contrast). In *Dammel, Antje; Kürschner, Sebastian; Nübling, Damaris (Hrsg.): Kontrastive Germanistische Linguistik*, volume 206-209 of *Germanistische Linguistik*, pages 827–862. Olms, Hildesheim.

Gary Libben, Monika Boniecki, Marlies Martha, Karin Mittermann, Katharina Korecky-Kröll, and Wolfgang U. Dressler. 2009. Interfixation in German compounds: What factors govern acceptability judgements? *Rivista di Linguistica*, 21(1):149–180.

Martin Neef and Susanne R. Borgwaldt. 2012. *Fugenelemente in neugebildeten Nominalkomposita*, pages 27–56. De Gruyter, Berlin, Boston.

Damaris Nübling and Renata Szczepaniak. 2008. On the way from morphology to phonology: German linking elements and the role of the phonological word. *Morphology*, 18(1):1–25.

Damaris Nübling and Renata Szczepaniak. 2013. Linking elements in German Origin, Change, Functionalization. *Morphology*, 23(1):67–89.

Lorelies Ortner and Elgin Müller-Bollhagen. 1991. *Hauptteil 4 Substantivkomposita*. De Gruyter, Berlin, Boston.

Roland Schäfer. 2018. *Einführung in die grammatische Beschreibung des Deutschen*. Number 2 in Textbooks in Language Sciences. Language Science Press, Berlin.

Roland Schäfer and Elizabeth Pankratz. 2018. The plural interpretability of German linking elements. *Morphology*, 28(4):325–358.

Carmen Scherer. 2012. *Vom Reisezentrum zum Reise Zentrum. Variation in der Schreibung von N+N-Komposita*, pages 57–82. De Gruyter, Berlin, Boston.

Barbara Schlücker. 2023. Compounding and Linking Elements in Germanic.

Barbara Schlücker. 2012. *Die deutsche Kompositionsfreudigkeit. Übersicht und Einführung*, pages 1–26. De Gruyter, Berlin, Boston.

Helmut Schmid, Arne Fitschen, and Ulrich Heid. 2004. SMOR: A German Computational Morphology Covering Derivation, Composition, and Inflection. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal. European Language Resources Association (ELRA).

Heide Wegener. 2003. Entstehung und Funktion der Fugenelemente im Deutschen, oder: warum wir keine Autosbahn haben. *Linguistische Berichte*, 196:425–457.

Heide Wegener. 2005. Das Hühnerei vor der Hundehütte : von der Notwendigkeit historischen Wissens in der Grammatikographie des Deutschen.

10. Language Resource References

Baayen, R. H. and Piepenbrock, R. and Gulikers, L. 1995. *CELEX2*. Linguistic Data Consortium, ISLRN 204-698-863-053-1. Web Download.

IDS. 2026. *Deutsches Referenzkorpus / Archiv der Korpora geschriebener Gegenwartssprache 2026-I*. Leibniz-Institut für Deutsche Sprache. PID https://www.ids-mannheim.de/dereko. Release vom 19.01.2026.

Schäfer, Roland and Pankratz, Elizabeth. 2018. *Dataset: The plural interpretability of German linking elements ("Morphology")*. Zenodo. PID https://doi.org/10.5281/zenodo.1323211.

A. Sources of Constraint Definitions

Source	Addressed Constraints
Eisenberg (2013, chapter 6.2)	(1), (3), (6), (7), (8), (10), (11), (23), (25), (27), (33), (34), (35), (55), (59), (60)
Fuhrhop and Kürschner (2015)	(1), (3), (23), (33), (40), (41), (42)
Koliopoulou (2014)	(23), (27), (36)
Kopf (2017)	(1), (27), (40), (48)
Krott et al. (2007)	(1), (16), (23), (33), (34), (64)
Kürschner	(1), (23), (25), (27), (29), (40), (42), (48)
Kürschner (2010)	(3), (23), (27), (28), (32), (40), (43), (44), (45), (47), (48), (51), (52), (53), (55), (56), (60), (61)
Libben et al. (2009)	(1), (33)
Neef and Borgwaldt (2012)	(36)
Nübling and Szczepaniak (2008)	(1), (6), (23), (27), (35), (38), (39), (41), (42), (48), (51), (55), (59), (60), (63)
Nübling and Szczepaniak (2013)	(3), (6), (9), (10), (13), (16), (17), (23), (25), (27), (31), (34), (35), (37), (41), (43), (47), (48), (49), (51), (55), (60), (62)
Ortner and Müller-Bollhagen (1991, chapter A.5)	(1), (2), (3), (4), (5), (7), (8), (10), (11), (12), (14), (16), (17), (18), (19), (23), (24), (25), (26), (27), (28), (30), (31), (32), (33), (34), (35), (36), (38), (39), (46), (49), (50), (51), (53), (54), (55), (57), (58), (62)
(Schäfer, 2018, chapter 8.1)	(3), (16), (23), (35), (64)
Schäfer and Pankratz (2018)	(1), (5), (10), (13), (15), (16), (23), (29), (33), (42), (43), (60), (62)
Schlücker (2012)	(1)
Schlücker (2023)	(23), (38), (64)
Wegener (2003)	(23), (42)
Wegener (2005)	(28), (38), (42), (47), (64)

Table 2: Linguistics literature providing the evidence for the nominal compounding constraints in German.

B. Calculation of Applicability Statistics: Example on Constraint (22)

Constraint (22) is potentially applicable to 2798 compounds formed from 32 unique first constituents. GeCoDB_v06 contains 282768 unique compounds built with 3613 unique first constituents. Then, type coverage of cov_{type} of (22) equals

$$\text{cov}_{type} = \frac{2798}{282768} \approx 0.01,$$

and item coverage cov_{item} of (22)

$$\text{cov}_{item} = \frac{32}{3613} \approx 0.009.$$

Constraint (22) is actually applicable to 2384 compounds. Therefore, type regularity reg_{type} of (22) is calculated as

$$\text{reg}_{type} = \frac{2384}{2798} \approx 0.852.$$

Thirty-one out of 32 first constituents always build compounds that conform to (22), so their group regularity equals 1.0. The remaining first constituent Liebe '*love*' builds 434 compounds in total, from which only 20 conform to (22), estimating Liebe's group regularity at $\frac{20}{434} \approx 0.046$. Then, item regularity reg_{item} of (22) amounts to

$$\text{reg}_{item} = \frac{31 \times 1.0 + 0.046}{32} \approx 0.97.$$

C. Coverage and Regularity Values: P2L vs. L2P

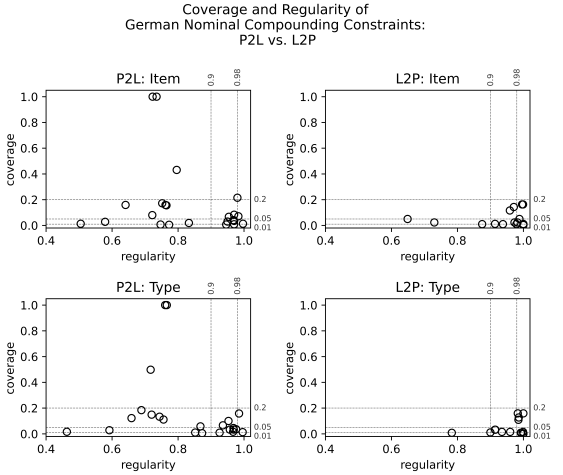


Figure 3: Distribution of the obtained coverage and regularity values with respect to the P2l/L2P type. Each P2l/L2P constraint appears twice: on the top and on the bottom.

D. Constraint Space Plot

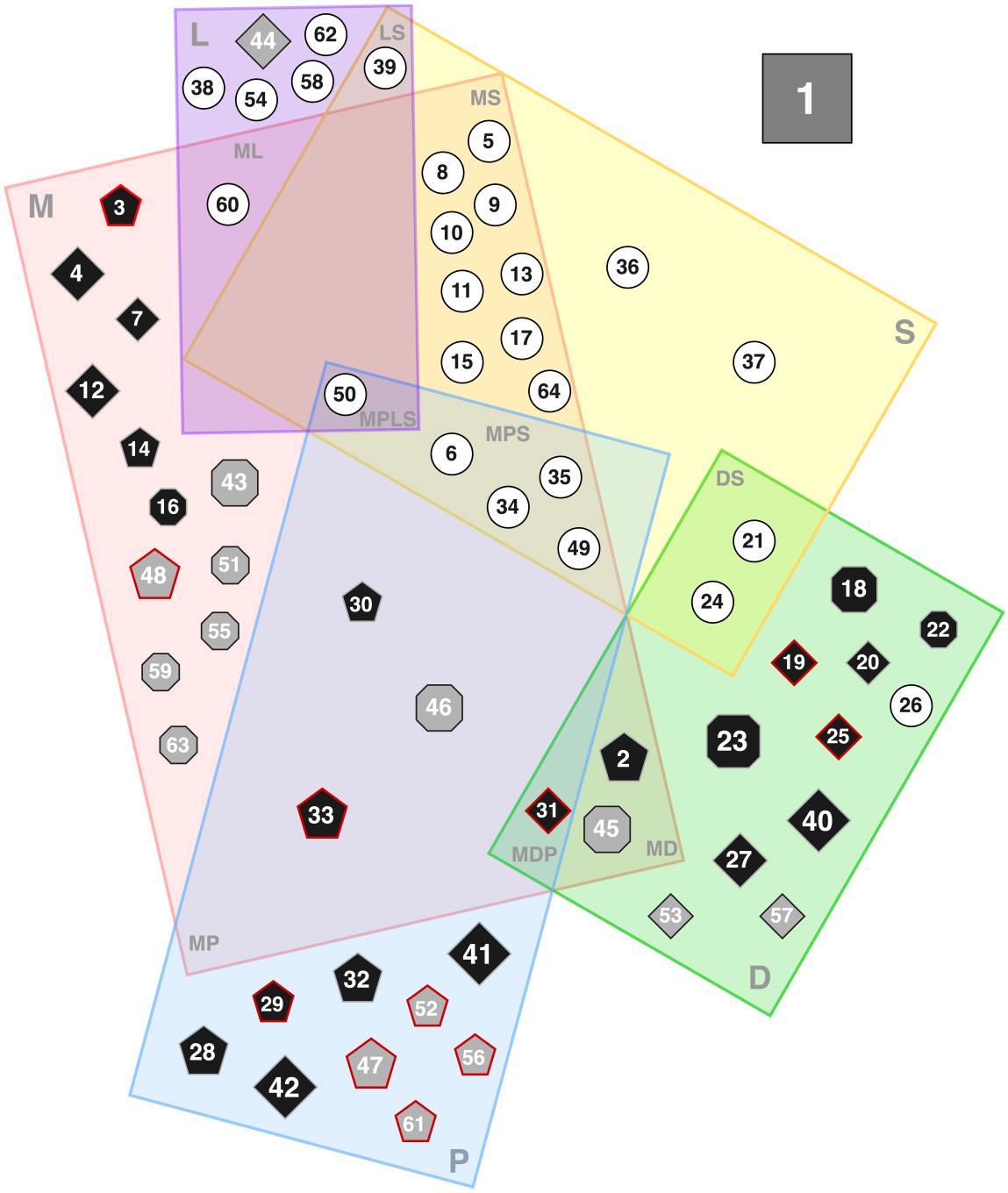


Figure 4: Overall illustration of the constraint space. Constraint (1) is the default constraint; it is placed separately outside of the box. For the other constraints, the size of the shape indicates the magnitude of its item coverage: small shapes — < 0.05, mid-sized shapes — in range $(0.05; 0.2)$, big shapes — ≥ 0.2. The number of angles in the shape is associated with the magnitude of item regularity: rhombs designate constraints with low and moderate regularity level, pentagons — increased regularity level, and octagons — high regularity level. Black shapes indicate P2L, and grey — L2P constraints. A red border signalizes that the constraint is partially/not supported by the empirical data. White circles represent constraints that could not be processed automatically. Large rectangles of different colors designate groups of constraints by the type of linguistic information they exploit: red is the morphological type, green is derivational, blue — phonological, yellow — semantic, and purple represents the lexical type. Mixed type groups are situated on the overlapping areas of the corresponding types. Each non-empty area is supplied with an abbreviation of the (mixed) type it represents. For the corresponding constraint definitions and the precise statistics values see Appendix E.

E. Constraint Definitions & Properties

Below is the list of German nominal compounding constraints summarized from the relevant literature. Each entry starts with the title and the definition of the corresponding constraint. The quantifiers are underlined. The grey tags correspond to the P2L/L2P type of the constraint, type of the constraint by employed linguistic information (abbreviated according to figure D), and the target item of the constraint (except for the default constraint (1)). For the 39 constraints selected for automated processing, item and type coverage and regularity are provided in form of a table, and the support level is specified under it. Examples, counterexamples (translations omitted), and the relevant evidence from the linguistics literature concludes the entry. The entries are separated by horizontal bars.

Constraint (1)

The majority of compounds have a zero linker. Zero linker is default for German compounds.

default | first + second constituent

	item	type
coverage	1.0	1.0
regularity	0.578	0.578

Support level: *supported*

Examples: Landøkreis, Hausøtür, Stadtømauer
Counterexamples: Land**es**regiesrung, H**ä**us**er**block, St**ä**dt**e**tag

Sources: Eisenberg (2013, 226), Fuhrhop and Kürschner (2015, 569), Kopf (2017, 177), Krott et al. (2007, 3), Kürschner, 107, Libben et al. (2009, 150, 156), Nübling and Szczepaniak (2008, 2), Ortner and Müller-Bollhagen (1991, 50, 52, 54, 103), Schäfer and Pankratz (2018, 8, 9), Schlücker (2012, 9)

Constraint (2)

First constituents that are constituted by simplex masculine or neuter nouns that build the plural form with a zero ending and simplex or complex feminine nouns that build the plural form with a zero ending attach *-ø-* almost regularly.

P2L | MD | first constituent

	item	type
coverage	0.067	0.066
regularity	0.954	0.936

Support level: *supported*

Examples: Schlittenøfahrt, Wagenørad, Uferøpromenade
Counterexamples: Orden**s**bruder, Teufel**s**werk, Frieden**s**angebot

Sources: Ortner and Müller-Bollhagen (1991, 106)

Constraint (3)

First constituents that are constituted by nouns that build the plural form with *-s* attach *-ø-* regularly.

P2L | M | first constituent

	item	type
coverage	0.035	0.033
regularity	0.97	0.957

Support level: *partially supported*

Examples: Balkonømöbel, Hoteløkette, Bobøbahn
Counterexamples: Uhu**s**nest, Deck**s**balken, Interim**s**phase, Karneval**s**kostüm

Sources: Eisenberg (2013, 227), Fuhrhop and Kürschner (2015, 572), Kürschner (2010, 9), Nübling and Szczepaniak (2013, 78), Ortner and Müller-Bollhagen (1991, 83, 106), Schäfer (2018, 229)

Constraint (4)

First constituents that are constituted by nouns that build the plural form with *-e* mostly have *-ø-*.

P2L | M | first constituent

	item	type
coverage	0.159	0.184
regularity	0.762	0.689

Support level: *supported*

Examples: Tischødecke, Projektøgruppe, Pilzøsammler
Counterexamples: Tag**e**gericht, Projekt**e**macherei, Hund**e**leine

Sources: Ortner and Müller-Bollhagen (1991, 83)

Constraint (5)

First constituents that are constituted by nouns that build the plural form with *-e* attach *-e-* irregularly in compounds in which the second constituent forces a plural meaning of the given first constituent.

P2L | MS | first constituent

Examples: Projekt**e**macherei, Tag**e**buch, Punkt**e**stand
Counterexamples: Pilzøsammler, Brotøkorb, Jahrøzehnt, Tag**e**lohn

Sources: Ortner and Müller-Bollhagen (1991, 103), Schäfer and Pankratz (2018, 29)

Constraint (6)

First constituents that are constituted by monosyllabic animal designations that build the plural form with *-e* attach *-e-* irregularly.

P2L MPS first constituent

Examples: Hund**e**leine, Swein**e**stall, Pferd**e**wagen
Counterexamples: Hund**s**stern, Schwein**ø**kram, Schaff**ø**stall

Sources: Eisenberg (2013, 228), Nübling and Szczepaniak (2008, 11), Nübling and Szczepaniak (2013, 81)

Constraint (7)

First constituents that are constituted by nouns that build the plural form with *-(")er* mostly attach *-ø-* or *-(")er-*.

P2L M first constituent

	item	type
coverage	0.019	0.058
regularity	0.833	0.868

Support level: *supported*

Examples: Buch**ø**deckel, B**ü**ch**er**regal, Kind**er**jacke
Counterexamples: Kind**es**alter, Mann**s**volk, Rind**ø**fleisch

Sources: Eisenberg (2013, 229), Ortner and Müller-Bollhagen (1991, 99)

Constraint (8)

First constituents that are constituted by nouns that build the plural form with *-(")er* attach *-ø-* in a majority of compounds in which the second constituent forces a singular meaning of the given first constituent.

P2L MS first constituent

Examples: Buch**ø**deckel, Grab**ø**stein, Blatt**ø**fläche
Counterexamples: M**ä**nn**er**herz, Kind**er**bild, Buch**ø**handel

Sources: Eisenberg (2013, 229), Ortner and Müller-Bollhagen (1991, 99, 108)

Constraint (9)

First constituents that are constituted by nouns that build the plural form with *-(")er* tend to attach *-ø-* in compounds in which the second constituent forces a mass meaning of the given first constituent.

P2L MS first constituent

Examples: Glas**ø**auge, Kraut**ø**salat, Horn**ø**haut
Counterexamples: Kraut**ø**garten

Sources: Nübling and Szczepaniak (2013, 81)

Constraint (10)

First constituents that are constituted by nouns that build the plural form with *-(")er* attach *-(")er-* more likely in compounds in which the second constituent forces a plural meaning of the given first constituent.

P2L MS first constituent

Examples: B**ü**ch**er**regal, Gr**ä**b**er**feld, Kr**ä**ut**er**frau
Counterexamples: Buch**ø**handel, Bild**ø**band, Kind**er**jacke, M**ä**nn**er**herz

Sources: Eisenberg (2013, 229), Nübling and Szczepaniak (2013, 80, 81), Ortner and Müller-Bollhagen (1991, 99), Schäfer and Pankratz (2018, 29)

Constraint (11)

First constituents that are constituted by person and animal designations that build the plural form with *-(")er* may attach *-(")er-* regardless of the singular/-plural interpretation of the given first constituent within the compound.

P2L MS first constituent

Examples: Kind**er**jacke, H**ü**hn**er**ei, Rind**er**brust

Sources: Eisenberg (2013, 229), Ortner and Müller-Bollhagen (1991, 98)

Constraint (12)

First constituents that are constituted by nouns that build the plural form with *-e* and umlaut mostly have *-ø-*.

P2L M first constituent

	item	type
coverage	0.08	0.111
regularity	0.722	0.756

Support level: *supported*

Examples: Hand**ø**fläche, Frucht**ø**joghurt, Arzt**ø**praxis
Counterexamples: H**ä**nd**e**druck, **Ä**rzt**e**streik, Fr**ü**cht**e**tee

Sources: Ortner and Müller-Bollhagen (1991, 83, 107)

Constraint (13)

First constituents that are constituted by nouns that build the plural form with *-e* and umlaut attach *-"e-* irregularly in compounds in which the second constituent forces a plural meaning of the given first constituent.

P2L MS first constituent

Examples: H**ä**nd**e**druck, **Ä**rzt**e**streik, G**ä**st**e**buch
Counterexamples: Baum**ø**gruppe, Zug**ø**verkehr

Sources: Nübling and Szczepaniak (2013, 81), Schäfer and Pankratz (2018, 29)

Constraint (14)

First constituents that are constituted by nouns that build the plural form with a zero ending and umlaut almost always have *-ø-*.

P2L | M | first constituent

	item	type
coverage	0.007	0.016
regularity	0.946	0.967

Support level: *supported*

Examples: Apfelø**b**aum, Klosterø**k**irche, Mutterø**s**prache
Counterexamples: Br**ü**derø**g**emeinde, M**ü**tterø**z**entrum, V**ä**terø**a**ufbruch

Sources: Ortner and Müller-Bollhagen (1991, 83, 106)

Constraint (15)

First constituents that are constituted by nouns that build the plural form with a zero ending and umlaut attach *-ø-* with umlaut irregularly in compounds in which the second constituent forces a plural meaning of the given first constituent.

P2L | MS | first constituent

Examples: Br**ü**derø**g**emeinde, M**ü**tterø**z**entrum, V**ä**terø**a**ufbruch
Counterexamples: Vogelø**f**utter

Sources: Schäfer and Pankratz (2018, 29)

Constraint (16)

First constituents that are constituted by mixed masculine and neuter nouns almost always attach *-s-*, *-ø-*, or *-(e)n-*.

P2L | M | first constituent

Examples: Staat**s**amt, Staat**en**bund, Bettø**a**nzug, Auge**n**lied
Counterexamples: Schmerz**ens**schrei

Sources: Krott et al. (2007, 3), Nübling and Szczepaniak (2013, 80), Ortner and Müller-Bollhagen (1991, 92), Schäfer (2018, 229), Schäfer and Pankratz (2018, 9)

Constraint (17)

First constituents that are constituted by mixed masculine and neuter nouns attach *-(e)n-* more likely in compounds in which the second constituent forces a plural meaning of the given first constituent.

P2L | MS | first constituent

Examples: Staat**en**bund, Bett**en**zahl, Strahl**en**belastung
Counterexamples: Motor**en**geräusch, Profes-sor**en**titel, Auge**n**lied, Interesse**n**bereich

Sources: Nübling and Szczepaniak (2013, 80), Ortner and Müller-Bollhagen (1991, 92)

Constraint (18)

First constituents that are constituted by derived nouns with suffixes *-er*, *-ler*, *-ner*, *-el*, *-sel*, *-chen*, *-lein* that build the plural form with a zero ending attach a zero linking element regularly.

P2L | D | first constituent

	item	type
coverage	0.071	0.045
regularity	0.983	0.968

Support level: *supported*

Examples: Reiterøschwert, Gürteløschnalle, Brötchenøgeber
Counterexamples: Alter**s**abstand, Reiter**s**mann

Sources: Ortner and Müller-Bollhagen (1991, 56, 82, 106)

Constraint (19)

First constituents that are constituted by derived nouns with suffixes *-bold*, *-nis*, *-rich*, *-at*, *-al* that build the plural form with *-e* attach *-ø-* regularly.

P2L | D | first constituent

	item	type
coverage	0.007	0.005
regularity	0.747	0.873

Support level: *not supported*

Examples: Erlaubnisøschein, Formatøvorlage, Personaløausweis
Counterexamples: Radikal**e**fänger, Inter-nat**s**leitung, Magistrat**s**verfassung

Sources: Ortner and Müller-Bollhagen (1991, 83, 107)

Constraint (20)

First constituents that are constituted by deverbal feminine nouns with a schwa suffix mostly attach *-ø-*.

P2L | D | first constituent

	item	type
coverage	0.028	0.028
regularity	0.579	0.592

Support level: *supported*

Examples: Abgabeøsoll, Sorgeøpflicht, Lageøplan
Counterexamples: Abgabe**n**ordnung, Anzeige**n**blatt, Lüge**n**detektor

Sources: Ortner and Müller-Bollhagen (1991, 93)

Constraint (21)

First constituents that are constituted by deverbal feminine nouns with a schwa suffix tend to attach *-(e)n-* in compounds in which the second constituent forces a plural reading of the given first constituent.

P2L DS first constituent

Examples: Abgabe**n**ordnung, Anzeige**n**blatt, Lüge**n**detektor
Counterexamples: Einnahme**ø**buch, Anzeige**ø**tafel

Sources: Ortner and Müller-Bollhagen (1991, 93)

Constraint (22)

First constituents that are constituted by deadjectival feminine nouns with a schwa suffix almost always attach *-(e)n-* or *-ø-*. Each of the two linkers is preferred in about the same number of cases.

P2L D first constituent

	item	type
coverage	0.009	0.01
regularity	0.97	0.852

Support level: *supported*

Examples: Fläche**ø**maß, Hitze**ø**welle, Tiefe**n**meter, Größe**n**klasse
Counterexamples: Liebe**s**brief

Sources: Fuhrhop and Kürschner (2015, 573), Kürschner, 112, Ortner and Müller-Bollhagen (1991, 93)

Constraint (23)

First constituents that are constituted by derived nouns with suffixes *-(ig)keit*, *-heit*, *-schaft*, *-ung*, *-sal*, *-ing*, *-ling*, *-tum*, *-um*, *-ion*, also *-ität* and its allomorphs attach *-s-* regularly.

P2L D first constituent

	item	type
coverage	0.215	0.158
regularity	0.98	0.985

Support level: *supported*

Examples: Gesundheit**s**amt, Gesellschaft**s**politik, Schicksal**s**drama, Rarität**s**wert, Eigentum**s**recht
Counterexamples: Minderheit**en**recht, Kuriosität**en**museum, Nationalität**en**kampf

Sources: Eisenberg (2013, 230), Fuhrhop and Kürschner (2015, 572), Koliopoulou (2014, 61), Krott et al. (2007, 3), Kürschner, 107, 115, Kürschner (2010, 20, 21), Nübling and Szczepaniak (2008, 10, 20, 23), Nübling and Szczepaniak (2013, 77), Ortner and Müller-Bollhagen (1991, 73, 83, 88, 89, 94), Schäfer (2018, 229), Schäfer and Pankratz (2018, 8), Schlücker (2023, 19), Wegener (2003, 448)

Constraint (24)

First constituents that are constituted by derived nouns with suffix *-ität* and its allomorphs prefer *-(e)n-* in compounds in which the second constituent forces a plural meaning of the given first constituent.

P2L DS first constituent

Examples: Rarität**en**sammlung, Kuriosität**en**händler, Aktualitäten**en**kino

Sources: Ortner and Müller-Bollhagen (1991, 94)

Constraint (25)

First constituents that are constituted by deverbal nouns that end in suffix *-en* attach *-s-* regularly.

P2L D first constituent

	item	type
coverage	0.005	0.009
regularity	0.773	0.926

Support level: *not supported*

Examples: Leben**s**mittel, Essen**s**gewohnheit, Unternehmen**s**bereich
Counterexamples: Essen**ø**portion, Leben**ø**geschäft, Husten**ø**tropfen, Beben**ø**gebiet

Sources: Eisenberg (2013, 230), Kürschner, 107, Nübling and Szczepaniak (2013, 78), Ortner and Müller-Bollhagen (1991, 89)

Constraint (26)

First constituents that are constituted by derived feminine nouns with suffix *-in* always attach *-(e)n-*. (The suffix -in adjusts orthographically in this case and becomes -inn.)

P2L D first constituent

Examples: Lehrerinn**en**mentalität, Freundinn**en**gruppe, Schülerinn**en**rat
Counterexamples: Königin**ø**mutter

Sources: Ortner and Müller-Bollhagen (1991, 94)

Constraint (27)

First constituents that are constituted by prefixed deverbal nouns exhibit a tendency to attach *-s-*.

P2L D first constituent

	item	type
coverage	0.159	0.122
regularity	0.641	0.659

Support level: *supported*

Examples: Anspruch**s**haltung, Eintrag**s**frist, Bedarf**s**fall, Verfall**s**datum

Counterexamples: Anruf**ø**beantworter, Überfall**ø**kommando, Bestand**ø**teil

Sources: Eisenberg (2013, 230), Koliopoulou (2014, 61), Kürschner, 107, Kürschner (2010, 21), Kopf (2017, 190), Nübling and Szczepaniak (2008, 18, 19), Nübling and Szczepaniak (2013, 78), Ortner and Müller-Bollhagen (1991, 89)

Constraint (28)

First constituents that are constituted by nouns that end in a sibilant or in a consonant cluster including [s] mostly adopt *-ø-*.

P2L | P | first constituent

	item	type
coverage	0.083	0.1
regularity	0.97	0.953

Support level: *supported*

Examples: Gefäß**ø**system, Fisch**ø**öl, Notiz**ø**buch, Herbst**ø**anfang
Counterexamples: G**ä**st**e**buch, Geist**er**haus

Sources: Kürschner (2010, 19), Ortner and Müller-Bollhagen (1991, 89, 109), Wegener (2005, 181)

Constraint (29)

First constituents that are constituted by nouns that end in a full vowel always adopt *-ø-*.

P2L | P | first constituent

	item	type
coverage	0.041	0.036
regularity	0.968	0.976

Support level: *partially supported*

Examples: Auto**ø**bahn, Uhu**ø**paar, Kuh**ø**milch, Gummi**ø**bär
Counterexamples: Uhu**s**nest, Idee**n**liste, Kalorie**n**wert, Schuh**e**kauf

Sources: Kürschner, 110, Schäfer and Pankratz (2018, 9)

Constraint (30)

First constituents that are constituted by feminine nouns that end with stressed *-ei*, *-ie*, *-ur*, also stressed or unstressed *-ik* usually attach *-ø-*.

P2L | MP | first constituent

	item	type
coverage	0.03	0.033
regularity	0.95	0.969

Support level: *supported*

Examples: Kosmetik**ø**industrie, Akustik**ø**paneele, Architektur**ø**büro, Metzgerei**ø**produkt
Counterexamples: Kultur**en**folge, Melodie**n**reigen, Partei**en**staat

Sources: Ortner and Müller-Bollhagen (1991, 107)

Constraint (31)

First constituents that are constituted by polysyllabic feminine nouns that end with [t] often attach *-s-* if the [t] is not part of a suffix *-(ig)keit*, *-heit*, *-schaft*, or *-ität* or its allomorphs.

P2L | MDP | first constituent

	item	type
coverage	0.013	0.016
regularity	0.505	0.463

Support level: *partially supported*

Examples: Arbeit**s**tag, Heirat**s**antrag, Zukunft**s**angst
Counterexamples: Arbeit**ø**geber, Jugend**ø**alter, Umwelt**ø**schutz

Sources: Nübling and Szczepaniak (2013, 77), Ortner and Müller-Bollhagen (1991, 74, 75)

Constraint (32)

First constituents that are constituted by nouns that end in schwa mostly adopt *-(e)n-*.

P2L | P | first constituent

	item	type
coverage	0.172	0.149
regularity	0.752	0.72

Support level: *supported*

Examples: Biene**n**zucht, Suppe**n**schüssel, Auge**n**lied
Counterexamples: Fläche**ø**maß, Sorge**ø**pflicht, Gebäude**ø**komplex, Gedanke**ns**turm

Sources: Kürschner (2010, 18), Ortner and Müller-Bollhagen (1991, 83, 92)

Constraint (33)

First constituents that are constituted by feminine nouns that end in schwa adopt *-(e)n-* regularly.

P2L | MP | first constituent

	item	type
coverage	0.156	0.133
regularity	0.765	0.744

Support level: *not supported*

Examples: Biene**n**zucht, Suppe**n**schüssel, Tiefe**n**meter
Counterexamples: Fläche**ø**maß, Sorge**ø**pflicht

Sources: Eisenberg (2013, 229), Fuhrhop and Kürschner (2015, 573), Krott et al. (2007, 3), Libben et al. (2009, 151, 157), Ortner and Müller-Bollhagen (1991, 83, 92), Schäfer and Pankratz (2018, 9)

Constraint (34)

First constituents that are constituted by consonant-final weak feminine nouns mostly adopt *-ø-* or *-s-* in compounds in which the second constituent forces a singular meaning of the given first constituent.

P2L | MPS | first constituent

Examples: Schriftøführer, Geburt**s**tag, Burgøanlage
Counterexamples: Burg**en**blick, Frau**en**bild

Sources: Eisenberg (2013, 229), Krott et al. (2007, 3), Nübling and Szczepaniak (2013, 80), Ortner and Müller-Bollhagen (1991, 94, 110)

Constraint (35)

First constituents that are constituted by consonant-final weak feminine nouns — especially final-stressed incl. monosyllabic ones — attach *-(e)n-* more likely in compounds in which the second constituent forces a plural meaning of the given first constituent.

P2L | MPS | first constituent

Examples: Schrift**en**verzeichnis, Geburt**en**kontrolle, Burg**en**land
Counterexamples: Burg**en**blick

Sources: Eisenberg (2013, 229), Nübling and Szczepaniak (2008, 3), Nübling and Szczepaniak (2013, 80), Ortner and Müller-Bollhagen (1991, 110), Schäfer (2018, 29)

Constraint (36)

Copulative compounds insert *-ø-* regularly. By copulative compounds are understood compounds in which the two constituents do not exhibit a clear modifier-head relation.

P2L | S | first + second constituent

Examples: Dichterøcomponist, Bettøsofa, Königinømutter

Sources: Koliopoulou (2014, 63, 64), Neef and Borgwaldt (2012, 31, 32), Ortner and Müller-Bollhagen (1991, 57, 91, 94, 109)

Constraint (37)

Argumental compounds are sometimes marked by *-s-*. By argumental compounds are understood compounds in which the second constituent still contains a high degree of verbiness and the first constituent of which constitutes their argument. The second constituent is such compounds is usually an agent designation, an *-ung* formation etc.

P2L | S | first + second constituent

Examples: Gewicht**s**heber, Krieg**s**führung, Unternehmen**s**beratung
Counterexamples: Gewichtøheber, Kriegøführung

Sources: Nübling and Szczepaniak (2013, 79)

Constraint (38)

Compounds belonging to technical terminology (economics, law, medicine, etc.) are often missing an explicit linking element.

P2L | L | first + second constituent

Examples: Erbschaftøsteuer, Herzøschlagen, Schadenøersatz
Counterexamples: Schaden**s**ersatz

Sources: Nübling and Szczepaniak (2008, 11), Ortner and Müller-Bollhagen (1991, 98), Schlücker (2023, 20), Wegener (2005, 177)

Constraint (39)

Compounds whose first constituent designates a person and whose second constituent is one of 'Mann', 'Frau', 'Leute', 'Tochter', 'Gattin', or 'Witwe' tend to insert *-s-*.

P2L | SL | first + second constituent

Examples: Reiter**s**mann, Lehrer**s**tochter, Bäcker**s**leute

Sources: Nübling and Szczepaniak (2008, 11), Ortner and Müller-Bollhagen (1991, 56)

Constraint (40)

The probability of *-s-* grows irregularly with the morphological complexity of the first constituent (any form of derivation).

P2L | D | first constituent

	item	type
coverage	1.0	1.0
regularity	0.735	0.766

Support level: *supported*

Examples: Gesundheit**s**amt, Gesellschaft**s**politik, Anspruch**s**haltung, Eintrag**s**frist
Counterexamples: Ort**s**amt, Krieg**s**ende, Anruføbeantworter, Minderheit**en**recht

Sources: Fuhrhop and Kürschner (2015, 572), Kopf (2017, 201), Kürschner, 106, 111, 112, Kürschner (2010, 19, 22)

Constraint (41)

The probability of *-s-* grows irregularly with the phonological complexity of the first constituent. By phonologically complex words are understood words with polysyllable non-trochaic form, words with unstressed prefixes, words with stressed or semi-stressed suffixes etc.

P2L P first constituent

	item	type
coverage	1.0	1.0
regularity	0.723	0.761

Support level: *supported*

Examples: Anruf**ø**beantworter, Beruf**s**erfahrung, Religion**s**unterricht
Counterexamples: Schlaf**ø**plan, Herbst**ø**anfang, Bestand**ø**teil

Sources: Fuhrhop and Kürschner (2015, 572), Nübling and Szczepaniak (2008, 10, 21, 22), Nübling and Szczepaniak (2013, 78)

Constraint (42)

The probability of *-s-* in compounds with simplex first constituents tends to decrease with the increasing sonority of the final segment of the first constituent. *-s-* is thus more frequent after plosives, infrequent after nasals and liquids, and it never occurs after a full vowel.

P2L P first constituent

	item	type
coverage	0.431	0.498
regularity	0.796	0.717

Support level: *supported*

Examples: Ort**s**tarif, Glück**s**rad, Himmel**ø**reich, Uhu**ø**paar
Counterexamples: Arbeit**ø**geber, Dieb**ø**stahl, Himmel**s**bahn, Uhu**s**nest

Sources: Fuhrhop and Kürschner (2015, 572), Kürschner, 110, Nübling and Szczepaniak (2008, 7, 17), Schäfer and Pankratz (2018, 9), Wegener (2003, 434), Wegener (2005, 180, 181, 182)

Constraint (43)

Almost all simplex nouns that constitute first constituents that attach *-s-* belong to a small fixed group of masculine and neuter nouns.

L2P M first constituent + linker

	item	type
coverage	0.05	0.032
regularity	0.988	0.915

Support level: *supported*

Examples: Zwang**s**jacke, Ort**s**angabe, König**s**haus
Counterexamples: Arbeit**s**amt, Heirat**s**antrag, Liebe**s**brief

Sources: Kürschner (2010, 23), Nübling and Szczepaniak (2013, 79), Schäfer and Pankratz (2018, 8)

Constraint (44)

Many simplex nouns that constitute first constituents that attach *-s-* have a high token frequency.

L2P L first constituent + linker

	item	type
coverage	0.05	0.032
regularity	0.649	0.914

Support level: *supported*

Examples: Volk**s**brauch, Amt**s**anwalt, Staat**s**amt
Counterexamples: Fasching**s**foto, Gral**s**sage, Konvent**s**sitzung

Sources: Kürschner (2010, 25)

Constraint (45)

All feminine nouns that constitute first constituents that attach *-s-* are morphologically complex.

L2P MD first constituent + linker

	item	type
coverage	0.163	0.159
regularity	0.996	0.983

Support level: *supported*

Examples: Gesundheit**s**amt, Gesellschaft**s**politik, Liebe**s**brief
Counterexamples: Arbeit**s**tag, Heirat**s**antrag

Sources: Kürschner (2010, 23, 24)

Constraint (46)

Almost all feminine nouns that constitute first constituents that attach *-s-* are polysyllabic.

L2P MP first constituent + linker

	item	type
coverage	0.163	0.159
regularity	0.999	1.0

Support level: *supported*

Examples: Gesundheit**s**amt, Gesellschaft**s**politik, Heirat**s**antrag, Liebe**s**brief

Sources: Ortner and Müller-Bollhagen (1991, 73)

Constraint (47)

All nouns that constitute first constituents that attach the *-n-* allomorph of the *-(e)n-* linker end in schwa.

L2P P first constituent + linker

	item	type
coverage	0.116	0.109
regularity	0.959	0.985

Support level: *partially supported*

Examples: Biene**n**zucht, Suppe**n**schüssel, Auge**n**lied
Counterexamples: Oper**n**haus, Elster**n**nest

Sources: Kürschner (2010, 18), Nübling and Szczepaniak (2013, 84), Wegener (2005, 179)

Constraint (48)

All nouns that constitute first constituents that attach *-(e)n-* belong to weak nouns.

L2P | M | first constituent + linker

	item	type
coverage	0.143	0.127
regularity	0.971	0.986

Support level: *partially supported*

Examples: Blume**n**topf, Katze**n**fell, Frau**en**hand
Counterexamples: Hahn**en**kamm, Mond**en**schein, Instrument**en**bau

Sources: Kopf (2017, 187), Kürschner, 118, 119, Kürschner (2010, 9), Nübling and Szczepaniak (2008, 3), Nübling and Szczepaniak (2013, 79, 84)

Constraint (49)

All masculine nouns that do not build the plural form with *-(e)n* that constitute first constituents that attach *-(e)n-* are monosyllabic designations of male persons, animals, astronomic objects, months.

L2P | MPS | first constituent + linker

Examples: Hahn**en**kamm, Mond**en**schein, Stern**en**himmel, Mai**en**nacht
Counterexamples: Mondøaufgang, Sternøbild, Maiøbaum

Sources: Nübling and Szczepaniak (2013, 79, 80), Ortner and Müller-Bollhagen (1991, 77, 78)

Constraint (50)

All neuter nouns that do not build the plural form with *-(e)n* that constitute first constituents that attach *-(e)n-* are polysyllabic foreign nouns that have a stressed final syllable that exhibit a plural meaning within the compound.

L2P | MPLS | first constituent + linker

Examples: Instrument**en**bau, Dokument**en**sammlung
Counterexamples: Zertifikatøsystem

Sources: Ortner and Müller-Bollhagen (1991, 78, 79)

Constraint (51)

All nouns that constitute first constituents that attach *-e-* build the plural form with *-e*.

L2P | M | first constituent + linker

	item	type
coverage	0.023	0.008
regularity	0.982	0.994

Support level: *supported*

Examples: Tag**e**buch, Punkt**e**stand, Hund**e**leine
Counterexamples: Herz**e**leid, Maus**e**zahn

Sources: Kürschner (2010, 9, 18), Nübling and Szczepaniak (2008, 5), Nübling and Szczepaniak (2013, 70, 81), Ortner and Müller-Bollhagen (1991, 102)

Constraint (52)

All nouns that constitute first constituents that attach *-e-* have a stressed last syllable.

L2P | P | first constituent + linker

	item	type
coverage	0.023	0.008
regularity	0.973	0.992

Support level: *partially supported*

Examples: Tag**e**buch, Punkt**e**stand, Hund**e**leine
Counterexamples: Anhalt**e**stelle

Sources: Kürschner (2010, 9, 18)

Constraint (53)

Most nouns that constitute first constituents that attach *-e-* are simplex.

L2P | D | first constituent + linker

	item	type
coverage	0.023	0.008
regularity	0.73	0.783

Support level: *supported*

Examples: Tag**e**buch, Punkt**e**stand, Hund**e**leine
Counterexamples: Anhalt**e**stelle

Sources: Kürschner (2010, 9, 18), Ortner and Müller-Bollhagen (1991, 102)

Constraint (54)

All nouns that constitute first constituents that attach *-e-* are native (none are loanwords).

L2P | L | first constituent + linker

Sources: Ortner and Müller-Bollhagen (1991, 102)

Constraint (55)

All nouns that constitute first constituents that attach *-(")er-* build the plural form with *-(")er*.

L2P M first constituent + linker

	item	type
coverage	0.01	0.015
regularity	0.979	0.999

Support level: *supported*

Examples: B**ü**ch**er**regal, Kr**ä**ut**er**frau, Kind**er**jacke

Sources: Eisenberg (2013, 229), Kürschner (2010, 9, 11, 18), Nübling and Szczepaniak (2008, 3), Nübling and Szczepaniak (2013, 80, 81), Ortner and Müller-Bollhagen (1991, 98)

Constraint (56)

All nouns that constitute first constituents that attach *-(")er-* have a stressed last syllable.

L2P P first constituent + linker

	item	type
coverage	0.01	0.015
regularity	0.938	0.96

Support level: *partially supported*

Examples: B**ü**ch**er**regal, Kr**ä**ut**er**frau, Kind**er**jacke
Counterexamples: Mitglied**er**liste, Vorbild**er**sammlung

Sources: Kürschner (2010, 9, 11)

Constraint (57)

Most nouns that constitute first constituents that attach *-(")er-* are simplex.

L2P D first constituent + linker

	item	type
coverage	0.01	0.015
regularity	0.875	0.935

Support level: *supported*

Examples: B**ü**ch**er**regal, Kr**ä**ut**er**frau, Kind**er**jacke
Counterexamples: Mitglied**er**liste, Vorbild**er**sammlung

Sources: Ortner and Müller-Bollhagen (1991, 98)

Constraint (58)

All nouns that constitute first constituents that attach *-(")er-* are native (none are loanwords).

L2P L first constituent + linker

Examples: B**ü**ch**er**regal, Kr**ä**ut**er**frau, Kind**er**jacke

Sources: Ortner and Müller-Bollhagen (1991, 98)

Constraint (59)

All nouns that constitute first constituents that attach *-"e-* build the plural form with *-e* and umlaut.

L2P M first constituent + linker

	item	type
coverage	0.011	0.003
regularity	1.0	1.0

Support level: *supported*

Examples: H**ä**nd**e**druck, **Ä**rzt**e**streik, G**ä**st**e**buch

Sources: Eisenberg (2013, 228), Nübling and Szczepaniak (2008, 5)

Constraint (60)

All nouns that constitute first constituents that attach *-es-* belong to a fixed row of masculine and neuter nouns and build the genitive form with *-(e)s-*. *-es-* is thus isolated.

L2P LM first constituent + linker

Examples: Bund**es**tag, Kind**es**alter, Tag**es**schau, Land**es**regierung

Sources: Eisenberg (2013, 229), Kürschner (2010, 11, 18), Nübling and Szczepaniak (2008, 3, 7), Nübling and Szczepaniak (2013, 81), Schäfer and Pankratz (2018, 8)

Constraint (61)

All nouns that constitute first constituents that attach *-es-* are monosyllabic.

L2P P first constituent + linker

	item	type
coverage	0.012	0.011
regularity	0.914	0.9

Support level: *partially supported*

Examples: Bund**es**tag, Kind**es**alter, Tag**es**schau, Land**es**regierung
Counterexamples: Gesetz**es**änderung, Vorjahr**es**winner, Bestand**es**aufnahme

Sources: Kürschner (2010, 11, 18)

Constraint (62)

All nouns that constitute first constituents that attach *-(e)ns-* belong to a fixed group of a few masculine nouns and a single neuter noun and have different genitive endings. *-(e)ns-* is thus isolated.

L2P L first constituent + linker

Examples: Schmerz**ens**geld, Name**ns**tag, Herz**ens**angst

Sources: Nübling and Szczepaniak (2013, 82), Ortner and Müller-Bollhagen (1991, 80, 97), Schäfer and Pankratz (2018, 9)

Constraint (63)

All nouns that constitute first constituents that attach *-ø-* with umlaut build the plural form with a zero ending and umlaut.

L2P | M | first constituent + linker

	item	type
coverage	0.003	0.0
regularity	1.0	1.0

Support level: *supported*

Examples: Br**ü**derøgemeinde, M**ü**tterøzentrum, V**ä**terøaufbruch

Sources: Nübling and Szczepaniak (2008, 5)

Constraint (64)

Linkers that are identical to the plural ending of the first constituent in a given compound tend to be associated with a plural meaning of this first constituent within the compound.

L2P | MS | first constituent + linker

Examples: Punkt**e**stand, B**ü**ch**er**regal, Staat**en**bund, Burg**en**land, M**ü**tterøzentrum
Counterexamples: Motor**en**geräusch, Burg**en**blick, Kind**er**jacke, Hund**e**leine, Pilzøsammler, Buchøhandel, Vogeløfutter

Sources: Krott et al. (2007, 3), Schäfer (2018, 230, 231), Schlücker (2023, 20), Wegener (2005, 173)

F. Suggestions to Corrections of Constraint Definitions

This appendix enlists our suggestions to corrections of constraint definitions that are partially/not supported by the empirical data. Dedicated Python checkers for the corrected versions of constraints proved that all the corrected versions are fully supported by the corpus data.

Constraint (3)

First constituents that are constituted by nouns that build the plural form with *-s* attach a zero linker regularly.

Correction: Almost all first constituents that are constituted by nouns that build the plural form with *-s* attach a zero linker.

	original	corrected
quantifier	regularly	almost all
item regularity	0.97	0.97

Constraint (19)

First constituents that are constituted by derived nouns with suffixes *-bold*, *-nis*, *-rich*, *-at*, *-al* that build the plural form with *-e* attach a zero linker regularly.

Correction: First constituents that are constituted by derived nouns with suffixes *-bold*, *-nis*, *-rich*, *-al* that build the plural form with *-e* attach a zero linker regularly.

	original	corrected
quantifier	regularly	regularly
item regularity	0.747	1.0

Constraint (25)

First constituents that are constituted by deverbal nouns that end in suffix *-en* attach *-s-* regularly.

Correction: First constituents that are constituted by deverbal nouns that end in suffix *-en* mostly attach *-s-*. Zero linker variants with a significantly lower frequency may occur.

	original	corrected
quantifier	regularly	mostly
item regularity	0.773	0.797

Constraint (29)

First constituents that are constituted by nouns that end in a full vowel always adopt a zero linker.

Correction: First constituents that are constituted by nouns that end in a full vowel adopt a zero linker regularly unless this noun is a loanword that ends with a stressed long [i] (*-ie*) or a stressed long [e] (*-ee*).

	original	corrected
quantifier	always	regularly
item regularity	0.968	0.993

Constraint (31)

First constituents that are constituted by polysyllabic feminine nouns that end with [t] often attach *-s-* if the [t] is not part of a suffix *-(ig)keit*, *-heit*, *-schaft*, or *-ität* or its allomorphs.

Correction: First constituents that are constituted by polysyllabic feminine nouns that end with [t] sometimes attach *-s-* if the [t] is not part of a suffix *-(ig)keit*, *-heit*, *-schaft*, or *-ität* or its allomorphs.

	original	corrected
quantifier	often	sometimes
item regularity	0.505	0.505

Constraint (33)

First constituents that are constituted by feminine nouns that end in schwa adopt -n- regularly.

Correction: <DISCARD AND MERGE WITH (32): even adding a simplex condition that eliminates deverbal and deadjectival nouns from consideration results in item regularity of 0.868; there is therefore no point in distinguishing between feminine and non-feminine nouns ending in schwa since both end up in the moderate regularity level.> First constituents that are constituted by nouns that end in schwa mostly adopt -n-.

	original	corrected
quantifier	regularly	mostly
item regularity	0.765	0.765

Constraint (47)

All nouns that constitute first constituents that attach the *-n-* allomorph of the *-(e)n-* linker end in schwa.

Correction: All nouns that constitute first constituents that attach the *-n-* allomorph of the *-(e)n-* linker end in schwa or in [ɑ] (unstressed *-er*).

	original	corrected
quantifier	all	all
item regularity	0.959	0.988

Constraint (48)

All nouns that constitute first constituents that attach *-(e)n-* belong are weak nouns.

Correction: Almost all nouns that constitute first constituents that attach *-(e)n-* belong are weak nouns.

	original	corrected
quantifier	all	almost all
item regularity	0.971	0.971

Constraint (52)

All nouns that constitute first constituents that attach *-e-* have a stressed last syllable.

Correction: All nouns that constitute first constituents that attach *-e-* have a stressed last syllable or are derived of those with a stressed prefix.

	original	corrected
quantifier	all	all
item regularity	0.973	1.0

Constraint (56)

All nouns that constitute first constituents that attach *-(")er* have a stressed last syllable.

Correction: All nouns that constitute first constituents that attach *-(")er* have a stressed last syllable or are derived of those with a stressed prefix.

	original	corrected
quantifier	all	all
item regularity	0.938	0.979

Constraint (61)

All nouns that constitute first constituents that attach *-es-* are monosyllabic.

Correction: All nouns that constitute first constituents that attach *-es-* are monosyllabic or are derived of those with a prefix.

	original	corrected
quantifier	all	all
item regularity	0.914	0.983

Additionally, we provide corrections for the constraints that were supported by the corpus data but appeared to be even more regular than specified in the relevant literature.

Constraint (16)

First constituents that are constituted by mixed masculine and neuter nouns almost always attach *-s-*, a zero linker, or *-(e)n-*.

Correction: First constituents that are constituted by mixed masculine and neuter nouns regularly attach *-s-*, a zero linker, or *-(e)n-*.

	original	corrected
quantifier	almost always	regularly
item regularity	0.997	0.997

Constraint (22)

First constituents that are constituted by deadjectival feminine nouns with a schwa suffix almost always attach -(e)n- or a zero linker. Each of the two linkers is preferred in about the same number of cases.

Correction: First constituents that are constituted by deadjectival feminine nouns with a schwa suffix always attach -(e)n- or a zero linker. Each of the two linkers is preferred in about the same number of cases.

	original	corrected
quantifier	almost always	always
item regularity	0.97	0.97

Constraint (28)

First constituents that are constituted by nouns that end in a sibilant or in a consonant cluster including [s] mostly adopt a zero linker.

Correction: First constituents that are constituted by nouns that end in a sibilant or in a consonant cluster including [s] adopt a zero linker almost regularly.

	original	corrected
quantifier	mostly	almost regularly
item regularity	0.97	0.97

Constraint (40)

The probability of *-s-* grows irregularly with the morphological complexity of the first constituent (any form of derivation).

Correction: The probability of *-s-* exhibits a tendency to grow with the morphological complexity of the first constituent (any form of derivation).

	original	corrected
quantifier	irregularly	tendency
item regularity	0.735	0.735

Constraint (41)

The probability of *-s-* grows irregularly with the phonological complexity of the first constituent. By phonologically complex words are understood words with polysyllable non-trochaic form, words with unstressed prefixes, words with stressed or semi-stressed suffixes etc.

Correction: The probability of *-s-* exhibits a tendency to grow with the phonological complexity of the first constituent. By phonologically complex words are understood words with polysyllable non-trochaic form, words with unstressed prefixes, words with stressed or semi-stressed suffixes etc.

	original	corrected
quantifier	irregularly	tendency
item regularity	0.723	0.723

Constraint (46)

Almost all feminine nouns that constitute first constituents that attach *-s-* are polysyllabic.

Correction: All feminine nouns that constitute first constituents that attach *-s-* are polysyllabic.

	original	corrected
quantifier	almost all	all
item regularity	0.999	0.999

Comparing Natural and Synthetic Structured Data: A Study of the Passive Verb Alternation in French and Italian

Giuseppe Samo[1], **Paola Merlo**[1,2]
[1]IDIAP Research Institute, [2]University of Geneva
{giuseppe.samo, paola.merlo}@idiap.ch

Abstract

This study compares the impact of natural and synthetic data on training and evaluating large language models (LLMs), using the case of passive verb alternation in French and Italian. We use Blackbird Language Matrices (BLMs), structured datasets designed to probe linguistic knowledge of underlying patterns across sentence sets. We compare structured templates instantiated with natural sentences extracted from Universal Dependencies to structured templates of synthetic sentences. Experiments show that while models achieve ceiling performance when trained and tested on synthetic datasets, they do not reliably generalize to natural sentences. In contrast, models trained on natural data exhibit robust performance across both natural and synthetic test suites, demonstrating their superior ability to capture abstract linguistic patterns. These results corroborate the value of natural data and of structured set ups in linguistic evaluation for probing LLMs' syntactic and semantic knowledge.

Keywords: Verb alternation, Universal Dependencies, Language Models, French, Italian

1. Introduction

Whether to use natural or synthetic data for evaluating linguistic knowledge has long been a central methodological debate in theoretical linguistics (Chomsky, 1965; Gibson and Fedorenko, 2013) and computational linguistics (cf. *test suites* vs. *test corpora*; Lehmann et al. 1996).

Natural data are easy to define: they are naturally occurring instances – 'wild data' (Bresnan, 2016) – extracted from corpora and can be annotated with various types of linguistic information. Synthetic data, by contrast, are artificially generated examples designed to target specific linguistic phenomena under controlled conditions. In this sense, synthetic datasets are conceptually akin to experimental materials in psycholinguistics, where carefully controlled stimuli are constructed to isolate particular grammatical factors (Schütze et al., 2013; Sprouse and Almeida, 2017; Futrell et al., 2019).

To ground this contrast, consider a simple verb alternation, where a verb can appear in a transitive or intransitive form (see also, unspecified object alternation, Levin 1993), such as the Italian verb *cantare* 'sing'. We can colour-code the core elements of the argument structure by marking the constituent with the Agent role in purple, the verb in teal, and the Theme in blue. In a synthetic setting, one can construct minimal, grammar-like examples that clearly instantiate the alternation and control the relevant variables. Consider, for example, the contrast between *La ragazza canta la canzone* (*the girl is singing the song*) and *La ragazza canta* (*the girl is singing*). The same structural configurations and their comparison, however, can also be retrieved from naturally occurring corpora by querying dependency relations such as `nsubj` and `obj` in annotated treebanks of Universal Dependencies (henceforth UD, Nivre 2015; De Marneffe et al. 2021). For example, consider the naturally occurring sentences below, which instantiate the same relevant constructions: '*Mina ha anche cantato molte canzoni di Mogol*' (*Mina [a singer] has sung many songs of [written by] Mogol [an author]* (ISDT, 2Parole_6-47)) and '*inoltre sino all'ultimo non si sapeva se il soprano ungherese Andrea Rost, ammalatasi nei giorni scorsi, avrebbe potuto cantare nel ruolo della protagonista Violetta, come alla fine è stato*' (*moreover, until the very end it was not known whether the Hungarian soprano Andrea Rost, who had fallen ill in the previous days, would be able to sing the role of the protagonist Violetta, as in the end she did* (ISDT, isst_tanl-1773)).

The notion of synthetic data has broadened considerably in recent years. It includes devised novel examples for experimental purposes, and it also refers to automatically generated outputs from language models and other conversational AI systems (Schepens et al., 2025; Civico, 2025). Such data may be synthetic in multiple senses: they are generated automatically rather than naturally occurring, and they can be tailored for experimental purposes. For example, Figure 1 shows an interaction in English with DeepSeek V.3. The interaction creates a synthetic instantiation for the alternation of the Italian verb *cantare* by prompting the generation of linguistic examples in English.[1] Recent work in NLP has increasingly explored the use of syn-

[1]For comparison, ChatGPT (GPT-5-mini, OpenAI) generated the following: "Intransitive use: *Ogni mattina Maria canta sotto la doccia.* (Here, cantare has no direct

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 39–51
May 11, 2026.

Human:
Can you make two sentences that show the fact that the verb *cantare* in Italian can be both transitive and intransitive?
DeepSeek:

> Certainly! Here are two sentences that demonstrate *cantare* used transitively (with a direct object) and intransitively (without a direct object):
> **Intransitive use:** Maria canta sotto la doccia ogni mattina. *(Here, cantare is used without a direct object—Maria is simply singing.)*
> **Transitive use:** Maria canta una bellissima canzone d'amore. *(Here, cantare takes a direct object—una bellissima canzone d'amore.)*

Figure 1: Example of a synthetic piece of data for transitive and intransitive forms of the Italian verb *cantare* 'sing' generated by DeepSeek (DeepSeek-AI, accessed Feb 13, 2026).

thetic data to complement or even replace natural data in training and evaluation pipelines (Goyal and Mahmoud, 2024; Josifoski et al., 2023; Chen et al., 2024). However, as noted by Jumelet et al. (2025), although large-scale annotated resources such as UD have become somewhat marginalized in the era of large language models (LLMs), the extensive linguistic knowledge encoded in these datasets continues to provide an essential foundation for rigorous and linguistically informed evaluation of current language models (Opitz et al., 2025).

This paper contributes to this debate by systematically comparing natural and synthetic data within a highly-structured diagnostic test suite targeting specific linguistic phenomena. We examine the extent to which each type of data helps language models identify and abstract the core linguistic elements underlying a complex phenomenon such as a verb alternation—one that cannot be adequately captured by considering sentences in isolation.

We construct structured datasets for the Blackbird Language Matrices (BLM) task (Merlo, 2023b,a; An et al., 2023; Samo et al., 2023; Nastase et al., 2024a,b; Jiang et al., 2024; Samo and Merlo, 2026a). The BLM task, inspired by Raven's Progressive Matrices (Raven, 1938), presents models with sentence-based puzzles designed to capture underlying linguistic rules. Specifically, each task instance consists of a context set of sentences that implicitly encodes the information required to complete a given linguistic pattern, and an answer set of minimally contrasting candidates, only one of which correctly replaces the missing element in the pattern defined by the context. This general template is then instantiated using carefully curated datasets. We instantiate a BLM template in two ways: with natural sentences extracted from UD and with synthetic sentences generated with the help of a ConversationalAI, each in two Romance languages, French and Italian.

The core linguistic phenomenon under investigation is the passive alternation (Levin, 1993, 85) because of its general applicability, across lexical items and languages. Unlike other verb alternations, the passive is not strongly tied to a specific lexical class, but most transitive verbs can undergo passivization (e.g. in Italian with the passive voice – colour coded in orange – as in *La canzone è cantanta dalla ragazza* 'the song is sung by the girl'), making it a structurally defined alternation, widely attested cross-linguistically (Siewierska, 2013).

Our contribution is two-fold: (i) we introduce multilingual, structured datasets that systematically compare natural and synthetic data with the BLM task and (ii) we evaluate how these data types may shape model representations and their capacity to generalize. This study sheds light on the interaction between training data for structured test suites and linguistic generalization, supporting Jumelet et al. (2025)'s intuition that natural data constitute a robust basis for evaluating language models.

2. Modelling the Passive Alternation with BLM Paradigms

Verb alternations have long been a central object of study in theoretical and computational linguistics, as they provide a window into how syntax, semantics, the lexicon, and the functional lexicon interact (Levin, 1993; Rappaport Hovav and Levin, 2024). However, many alternations that are commonly investigated are lexically specific, only restricted verb classes participate, and cross-linguistic correspondences are often unstable (Fillmore et al., 2003). This lexical specificity can favour superficial patterns in pretrained language models, which may capture surface-level lexical associations rather than abstract syntactic generalizations. The passive alternation, instead, is less restricted in the lexical nature of the verb it can apply to.

Passive constructions preserve the core argument structure while systematically altering the mapping between syntactic positions and discourse roles (Hopper and Thompson, 1980). The core argument structure of the verb is preserved, but its syntactic realization changes. The verb's Theme is

object—Maria is simply singing.); Transitive use: *Maria canta una canzone napoletana.* (Here, cantare takes a direct object—una canzone napoletana.). Accessed Feb 13, 2026.

promoted to subject position and, in languages with overt agreement, controls subject–verb agreement, while the Agent can be omitted from the surface form, yielding structures that are underlyingly diadic (like the transitive sentence in the introduction) but appear monadic on the surface (like the intransitive sentence in the introduction).

Across languages, passives are typically signaled by changes in word order and functional morphosyntactic markers. In Italian and French, passive constructions are mainly marked by the presence of an auxiliary verb. Italian typically uses *essere* 'be' or *venire* 'come', while French primarily uses *être* 'be' and, in certain constructions, also *faire* 'do'. However, these auxiliaries are not specific to passive structures. For example, in both languages *essere* and *être* 'be' also introduce copular and predicative constructions; in Italian, *essere* is additionally used to form compound tenses with certain classes of verbs such as unaccusatives (Burzio, 1986). In French, *faire* 'do' can also be used in causative constructions. Finally, *venire* 'come' – as well the other auxiliaries – function as lexical verbs that can occur in isolation.

The passive form is particularly relevant, as it constitutes a highly pervasive phenomenon in Romance, especially in specific text genres (Brunato et al., 2022), while at the same time displaying considerable structural complexity (Volpato, 2010) and giving rise to effects at the pragmatic level (Reinhart, 1981, 64). As a result, passivization is less fine-grained, and also differently generalizable, than other lexical alternations.

As a key methodological choice in our study of alternation phenomena, we move beyond isolated sentences as the primary unit of analysis. Single sentences are often insufficient, since alternations are inherently relational; they concern systematic correspondences between forms rather than properties of individual strings (unlike, for instance, agreement or long-distance dependencies). Minimal pairs are also a well-established tool in linguistic analysis, but they rely on tightly controlled contrasts and inevitably remain tied to specific lexical items.

For this reason, we adopt a paradigm-based perspective, where the primary object of analysis is a structured set of systematically related constructions. We chose to evaluate the knowledge of passive with the BLM task, which has been shown to be one of the most challenging evaluations for testing Italian knowledge in LLMs (Nissim et al., 2025).

The BLM template The template adopted in this study is shown in Figure 2. BLM templates are typically constructed around three core elements: external rules *E*, internal rules *I* and relational operators *R* (Merlo, 2023b). The external rule *E* of

Template			
Structure	R_{Arg}	R_{ST}	Example
CONTEXT SET			
1 Ag Vact Th ?	2	Q	*Does the customer pay the bill?*
2 Ag Vact Th	2	D	*The student gets the prize.*
3 Ag Vact ?	1	Q	*Does the teacher explain?*
4 Ag Vact	1	D	*The car moves.*
5 Th Vpass Ag ?	2	Q	*Why is the case studied by the lawyer?*
6 Th Vpass Ag	2	D	*The key is found by the boy.*
7 Th Vpass ?	1	Q	*When was the screen touched?*
8 ???			
ANSWER SET			
1 Th Vpass	1	D	***The plants were watered***
2 Th Vpass Ag	2	D	*The news is reported by the speaker.*
3 Ag Vact	1	D	*The writer publishes.*
4 Ag Vact Th	2	D	*The store ships the order.*
5 Question	1 or 2	Q	*How was the data analyzed?*

Figure 2: BLM template structure instantiated with a synthetic example in English, generated with DeepSeek V.3 (section 3.2). Ag = Agent, Th = Theme, Vact = verb in active voice, Vpass = verb in passive voice, red elements mark interrogative markers. Number of arguments (1, 2) and sentence type (Q = question, D = declarative).

the phenomenon that distinguishes passive from active voice concerns verbal morphology. The internal rule *I* relates to the syntax–semantic mapping of the verb's arguments. In passive constructions, the Theme is promoted to syntactic subject position and agrees with the verb. The *R* components specifies how the linguistic objects of the BLM are connected within the template. In the proposed template, *R* corresponds to sentence type (R_{ST}) and number of arguments superficially realized in the sentence (R_{Arg}). These two linguistic phenomena are external to the one under investigation and do not influence the learning of *E* and *I* rules. We explore the contrast between interrogative and declarative sentences, but also between bi-argumental and mono-argumental structures.

The inclusion of both declarative and interrogative clauses is partly motivated by the structure of the available treebanks and the distributions of sentences across treebanks. For example, the French QuestionBank (FQB, Seddah and Candito 2016) exclusively contains interrogative sentences, and allows the retrieval of a large number of instances for task training and testing purposes. Declaratives are also present in large scale in the UD treebanks of French and Italian (Samo and Merlo, 2021). From a linguistic perspective, sentence type constitutes an independent dimension with respect to passive voice alternation. Interrogative marking may affect word order, but does not interfere with the morphological marking of passive voice (*E*), nor with the argument-structure reconfiguration involved in the promotion of the Theme (*I*).

We therefore develop a template that manipulates the organization of the three core elements *E*,

I, and *R*. Elements of *E* include the verb inflected for the active voice (Vact) and for the passive voice (Vpass). The *I* component concerns the syntactic functions of the core arguments, Agents and Themes. The *R* component specifies whether both Agents and Themes are present in the same string (or only one of these arguments appears, R_{Arg}) and whether the sentence is interrogative or declarative. Interrogative sentences are marked typographically by the interrogative symbol ?, and also by the presence of *wh*-elements (R_{ST}).

Context Set Formally, the context set instantiates a 2 × 2 × 2 design crossing voice (active/passive), number of argument realizations (one or two overt arguments), and sentence type (interrogative vs. declarative). The eighth sentence is left uninstantiated. The organization of the context reflects two independent alternations built on the different *R*s (R_{Arg} and R_{ST}).

Each pair of consecutive sentences forms a minimal interrogative–declarative paradigm: in all four pairs, the first member is interrogative and the second is declarative. For every four sentences, the first two show two overt arguments, while the last two show only one argument. The resulting structure creates a double alternation pattern (and a triple one with active-passive): sentence type alternates within each pair, while voice and argument realization alternate across pairs.

The first pair (1–2) consists of active clauses with two overt arguments. The second pair (3–4) maintains the active voice, but in mono-argumental structures. The third pair (5–6) shifts to the passive voice while restoring the bi-argumental configuration, with the Theme promoted to subject position and the Agent realized as an overt agentive complement. Finally, the fourth pair is designed to instantiate mono-argumental passive clauses; however, only the interrogative configuration (7) is provided in the context. The corresponding declarative passive without an overt agent (8) is omitted and must be inferred. The missing final sentence can therefore be recovered only by integrating all the patterns simultaneously.

Answer Set The correct answer is the declarative passive sentence without an agentive prepositional phrase. The answer set is designed to cover all combinations of argument number (one or two arguments) and voice (active vs. passive), and an additional interrogative option. Therefore, the errors are of three types: (i) errors of VOICE when the verb is inflected in the active voice, violating *E* and implicitly *I*; (ii) errors of NUMBER OF ARGUMENTS when there are two arguments; (iii) errors of SENTENCE TYPE when the sentence is not a declarative, both violations of *R*. In our case, candidate answer 2 (Th Vpass Ag) is an error of NUMBER OF ARGUMENTS because it also shows an additional argument, the agentive PP; sentence 3 (Ag Vact) is an error of VOICE because it is a mono-argumental declarative, but not a passive structure; sentence 4 (Ag Vact Th) is an error of both VOICE and NUMBER OF ARGUMENTS; sentence 5 (question) is mainly an error of SENTENCE TYPE.

3. Natural and Synthetic Data for BLM

The template illustrated in Figure 2 is instantiated with natural and synthetic data. Examples are given in French and Italian respectively in Figure 3 and Figure 4.[2]

3.1. Natural data

All the sentences are extracted from UD treebanks. We retrieved natural occurring sentences with the system GREWMATCH (MATCH.GREW.FR) (Guillaume, 2021). For French, we extracted questions from the French QuestionBank v.2.17 (FQB, 23,345 tokens; Seddah and Candito 2016) containing only questions from news and nonfiction sources. We extracted declarative sentences from the French GSD data v.2.17 (389,364 tokens; Guillaume et al. 2019), which contains data from blogs, news, reviews and encyclopeadic entries. For Italian, we retrieved sentences from ISDT v.2.17 (278,424 tokens; Bosco et al. 2014) containing data from legal texts, news and Wikipedia and VIT v.2.17 (259,625 tokens; Alfieri and Tamburini 2016), which which consists of news and nonfiction texts.

The sentence types of the BLM template were directly instantiated using sentences that matched the corresponding structures in the treebanks. Each context and answer sentence was retrieved using simple queries. The Agent in the sentences was identified as the target of the `nsubj` dependency associated with a variable V in active sentences, and as the dependent of a `obl:agent` dependency in passive sentences. The Theme was identified as the dependent of `obj` in active sentences and of the `nsubj:pass` dependency in passive sentences. To retrieve questions, we added the query for an interrogative marker corresponding to a variable Q whose FORM was annotated as "?". We restricted our search to sentences – intended here as the output of an instance of the query – with a single verb by excluding output sentences containing additional verbs, using the constraint Y (a variable different than V) whose UPOS was "VERB".

[2]Data is available at https://www.idiap.ch/en/scientific-research/data/blm-passF and https://www.idiap.ch/en/scientific-research/data/blm-passI.

	Natural	Synthetic
	CONTEXT SET	
1	*Que signifie l'acronyme NASA ?* 'what does the acronym NASA mean?' (FQB_TREC-fr-584)	*Le garçon jette-t-il la pierre?* 'Does the boy throw the stone?'
2	*Leur album No Dice obtient un excellent accueil.* 'Their album *No Dice* obtained an excellent reception' (fr-ud-train_06225)	*L'équipe félicite le gagnant.* 'The team congratulates the winner'
3	*Combien de temps dure le voyage entre Tokyo et Niigata ?* 'How long does the journey from Tokyo to Niigata take?' (FQB_TREC-fr-111)	*L'équipe coûte-t-elle?* 'Does the team cost?'
4	*Bryon Anthony McCane naît en 1976 d'une mère italienne et d'un père afro-américain.* 'Bryon Anthony McCane was born in 1976 to an Italian mother and an African-American father.' (fr-ud-train_00556)	*Le chanteur chante.* 'The singer sings'
5	*Les moteurs rotatifs étaient produits par qui ?* 'By whom were rotating engines invented?' (FQB_TREC-fr-886)	*Comment la scène est-elle décrite par l'écrivain ?* 'How is the scene described by the writer?'
6	*Uber a été fondé par Garrett Camp, Travis Kalanick et Oscar Salazar en 2009.* 'Uber was founded by Garrett Camp, Travis Kalanick and Oscar Salazar in 2009' (fr-ud-train_02964)	*Un livre est écrit par l'auteur.* 'A book is written by the author'
7	*Dans quelle province française le cognac est-il produit ?* 'In which French province is cognac produced?' (FQB_TREC-fr-1134)	*Quand la musique a-t-elle été composée?* 'When was the music composed?'
8	???	???
	ANSWER SET	
1	***Le château est ensuite vendu plusieurs fois*** 'The castle is then sold several times' (fr-ud-train_00010)	***Les données ont été analysées.*** The data were analyzed
2	*Les travaux furent dirigés par le Florentin Girolamo della Robbia et les Tourangeaux Pierre Gadier et Gatien François.* 'The works were directed by the Florentine Girolamo della Robbia and the Turingians Pierre Gadier and Gatien François' (fr-ud-train_07752)	*La langue est apprise par l'étudiant.* 'The language is learned by the student.'
3	En tout cas je n'y retourne pas et ma collègue non plus ! 'In any case, I'm not going back there and neither my colleague!' (fr-ud-train_00110)	*Le programmeur code.* 'The programmer codes.'
4	*En mars 2010, il signe son premier contrat professionnel avec Birmingham City.* 'In March 2010, he signed his first professional contract with Birmingham City' (fr-ud-train_01336)	*Le parent paie la facture.* 'The parent pays the bill'
5	*Quel âge a le Soleil ?* 'How old is the Sun?'(FQB_TREC-fr-334)	*Quand le colis a-t-il été reçu?* 'When was the package received?'

Figure 3: Natural and synthetic examples in French, glosses and ID number (natural data) for reference within the explored treebanks. We have coloured code the core elements of the BLM-template. Correct answer in bold.

	Example Natural	Example Synthetic
	CONTEXT SET	
1	*Quando Innsbruck ospitò le Olimpiadi Invernali?* 'When did Innsbruck host the Winter Olympic Games?' (ISDT, quest-991)	*Lo scrittore finisce il romanzo?* 'Does the writer finish the novel?'
2	*I carabinieri gli hanno recapitato il decreto di revoca degli arresti domiciliari.* 'The Carabinieri delivered to him the order revoking his house arrest.' (ISDT, isst_tanl-280)	*Il giocatore segna un gol.* 'The player scores a goal.'
3	*A che velocità viaggia la luce?* 'At what speed does light travel?' (ISDT, quest-656)	*Il bambino disegna?* 'Does the child draw?'
4	*E nel frattempo politici, ambasciatori e industriali discutevano di petrolio.* 'And in the meantime politicians, ambassadors, and industrialists were discussing oil.' (ISDT, isst_tanl-186)	*L'editore corregge.* 'The publisher corrects.'
5	*Chi è stato sconfitto da Andrei Medvedev nella finale del Torneo di Monte Carlo ?* 'Who was defeated by Andrei Medvedev in the Monte Carlo tournament final?' (ISDT, quest-553)	*Quando sono annaffiate le piante dal giardiniere?* 'When are the plants watered by the gardener?'
6	*Le leggi sono promulgate dal Presidente della Repubblica entro un mese dall' approvazione* 'The laws are promulgated by the President of the Republic within one month of approval.' (ISDT, tut-1546)	*La squadra è tifata dal bambino.* 'The team is supported by the child.'
7	*Come può essere eliminato il rigetto del sistema immunitario?* 'How can immune system rejection be eliminated?' (quest-297)	*Quando è stato sollevato il peso?* 'When was the weight lifted?'
8	???	
	ANSWER SET	
1	***Dopo tre anni di discussioni la proposta non venne accolta.*** 'After three years of discussions, the proposal was not accepted.' (VIT-1304)	***Lo strumento è stato suonato.*** 'The instrument was played.'
2	*La lettera di trasporto aereo viene emessa dal mittente in tre esemplari originali.* 'The air waybill is issued by the sender in three original copies.' (ISDT, splet-161)	*L'email è inviata dal capo.* 'The email is sent by the boss.'
3	*La squadra nazionale italiana di calcio giocherà la prima partita del Campionato mondiale il 12 giugno, ad Hannover, con il Ghana.* 'The Italian national football team will play its first World Cup match on June 12 in Hanover against Ghana.' (ISDT, 2Parole_1-214)	*Il cassiere conta i soldi.* 'The cashier counts the money.'
4	*I disegni risentono delle influenze arabe e a volte anche di quelle francesi* 'The drawings show Arab influences and sometimes French ones as well.' (ISDT, isst_tanl-3212)	*Lo chef cucina.* 'The chef cooks.'
5	*Chi possiede la Saab?* 'Who owns the Saab?' (ISDT, quest-422)	*L'autista parcheggia l'auto?* 'The driver parks the car?'

Figure 4: Natural and synthetic examples in Italian, glosses and ID number (natural data) for reference within the explored treebanks. We have coloured code the core elements of the BLM-template. Correct answer in bold.

The combination of these factors led to the retrieval of natural instances for each sentence of the context and the answer set.[3]

Samples of the naturally extracted data were manually checked to assess whether the queries returned the relevant structures.

3.2. Synthetic data

We created our dataset using the DeepSeek-V3 model (Guo et al., 2025), adopting a hybrid approach that combines LLM generation with explicit linguistic constraints. This ensured the generated sentences remained grammatically well-formed and semantically coherent within each alternation paradigm. Specifically, the prompt instructed the conversationalAI to generate a structured linguistic dataset consisting of "100 simple sentences with frequent subjects, verbs and objects'" (5 interactions) in English and transform them into eight distinct syntactic variants corresponding to the context and answer set structures (interrogative, intransitive, and passive forms with and without agentive phrases), with specific formatting requirements for direct import into spreadsheet software. In later interactions, the model was required to translate the sentences into French and Italian. A native speaker of Italian and proficient in French and English, manually checked the translations. The intervention was minimal, as the sentences were generally well-formed.[4]

English was adopted as a pivot language (Wendler et al., 2024) to construct a single abstract dataset before its language-specific realization. This design choice ensures that the French and Italian versions do not stand in a derivational relationship to each other, but instead instantiate the same underlying synthetic templates. As a result, the synthetic status of the data is methodologically consistent across languages: neither dataset is more natural, or derived, than the other, since both originate from the same controlled generative procedure.

The sentences were specifically designed to isolate the factors under investigation: as shown in Figures 3 and 4, they included exclusively the elements relevant to the BLM (agents, themes, verbs in active and passive voice and interrogative markers).

[3]All queries are available in Table 1 in the Appendix.

[4]Out of 4,000 sentences (500 for each of the eight structures), only 43 were manually corrected for French (0.01) and 64 for Italian (0.02). The intervention primarily ensured that sentences remained unambiguously transitive or intransitive, removing verbs with reflexive-like elements when relevant for the template.

4. Experiments

The experiments systematically compare natural and synthetic data in a highly structured setup, the BLM test suite. The goal is to evaluate how much each type of data helps language models identify and abstract the core linguistic elements underlying the passive alternation. We explore the behavior of a simple probe through a series of experiments.

We adopt a feed-forward neural network (FFNN) architecture as described in Samo et al. (2023). For each sentence in the BLM, we compute an embedding by averaging its token representations obtained from pretrained models. The FFNN takes as input the concatenated embeddings representing the context, is trained using a max-margin loss objective, and predicts the answer whose embedding achieves the highest cosine similarity with the network's output. Following previous work (Nastase et al., 2025), we first test embeddings from monolingual ELECTRA models (French: *dbmdz/electra-base-french-europeana-cased-discriminator*; Italian: *dbmdz/electra-base-italian-xxl-cased-discriminator*) and then from a multilingual model (*google/electra-base-discriminator*). Monolingual and multilingual models may exhibit asymmetries in performance due to differences in their token representations (Samo and Merlo, 2026b).

We run both experiments when training and testing belong to the same data set type and also experiments that cross the two types. Each dataset, constituted of 2000 BLM instances, is split into separate training and testing sets (80%–20% split), ensuring no overlap of instances between them to control data leakage.

These settings are referred to as SynSyn (train and test on synthetic) and NatNat (train and test on natural). SynNat refers to training the model with synthetic data and testing its ability to generalize to natural instances, while NatSyn denotes training on BLM-templates instantiated with natural data and testing on its ability to abstract to synthetic instances.

4.1. Results: Monolingual

Figure 5 shows the results, measured in F1 scores, for the monolingual models. Among the monolingual models, the full synthetic suite achieves the highest performance (French 1.00; Italian 0.99), outperforming the natural suites (French 0.62; Italian 0.77).

When models are trained on synthetic data but tested on natural data, performance drops significantly, reaching near-chance levels (French 0.29; Italian 0.28). In contrast, training on natural data and testing on synthetic data yields performance comparable to that of the full natural suites, with

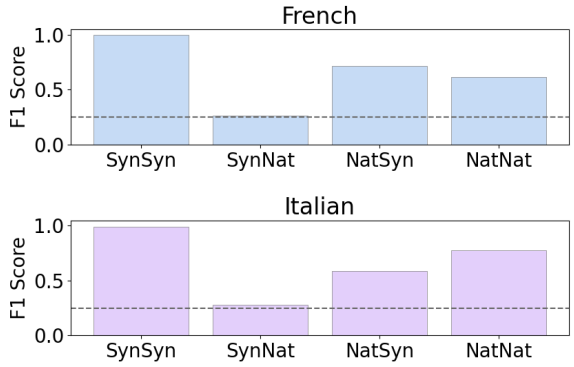


Figure 5: F1 scores across training and test suites in monolingual models. The grey dotted line indicates chance level.

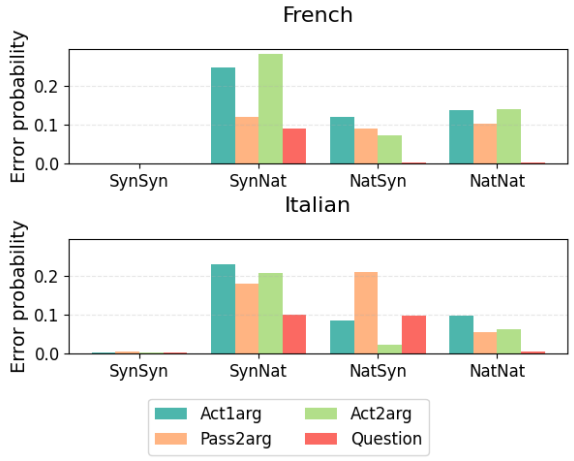


Figure 6: Error probabilities and types of errors (indicated by the selected erroneous answer) in monolingual models. Act = Active; Pass = Passive; arg = arguments.

French even showing improved performance with respect to natural test suites.

The error analysis is presented in Figure 6. Across all suites, errors related to SENTENCE TYPE (questions) are never the most prominent while the other errors are more evenly distributed. However, in the Italian NATSYN, the most prominent error type involves passive structures with two arguments, suggesting that voice was a learned feature.

4.2. Comparing Monolingual and Multilingual Embeddings

Figure 8 shows the results of the comparison between monolingual and multilingual models. The results follow the same pattern across languages and models. Multilingual models show competitive, and in some cases superior peformance. In particular, the multilingual model reaches ceiling performance in the NATSYN suite for French (0.99),

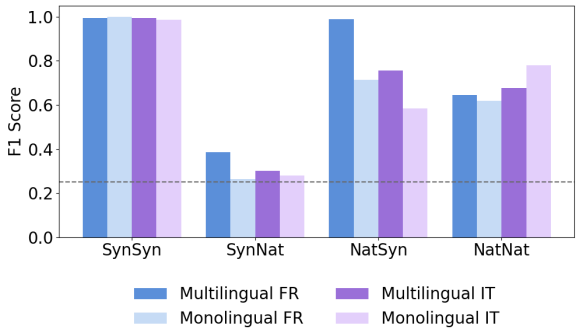


Figure 7: F1 scores across conditions, languages and models. The gray dotted line indicates chance level. IT = Italian, FR = French.

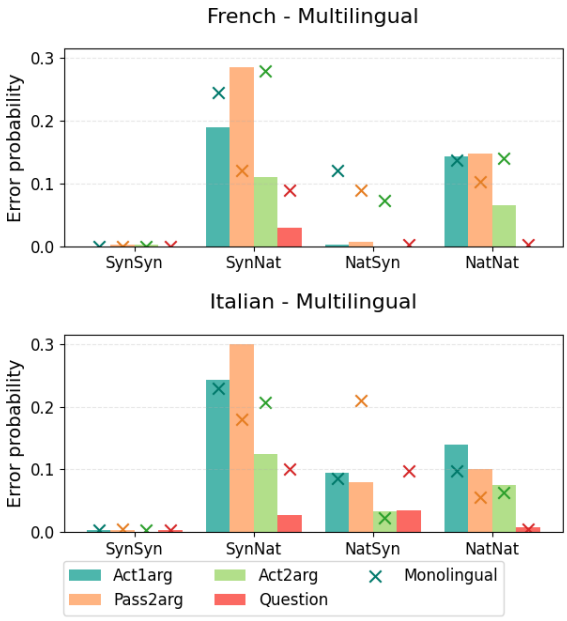


Figure 8: Error probabilities and types of errors (indicated by the selected erroneous answer) in multilingual models. Crosses repeat values of error bars in monolingual model for comparison (see Figure 6). Act=Active, Pass=Passive, arg=arguments.

indicating strong comprehension of the BLM and generalisation ability from natural to synthetic instances.

Consider the error analysis. In the NATNAT setting the most frequent incorrect responses in the multilingual models involve VOICE or NUMBER OF ARGUMENTS. This pattern indicates that active sentences with two arguments (i.e., violations of two rules) are not considered plausible answers. Errors related to SENTENCE TYPE are minimal in the multilingual models, even in the lowest-performing configuration. This suggests that the models consistently learn the sentence type encoded in the BLM template across languages and training con-

ditions.

However, multilingual and monolingual models differ in the type of error that is most prominent in the lowest-performing setting, SynNat. In both languages, the most frequent errors produced by the multilingual models are passive structures, suggesting that these models successfully learn verb voice from synthetic training data. In contrast, monolingual models show a higher proportion of active constructions as their most prominent error.

5. Discussion

Although the full synthetic suites outperform the full natural suites, overall performance drops significantly when models are trained on synthetic data but tested on natural data (aggregated, *t*-test: $t(6) = 27.07$, $SE = 0.025$). Although models trained on natural data show lower absolute performance, their results are more stable across different evaluation settings.

Despite the fact that models trained and tested on synthetic data achieve near-ceiling performance, this competence does not reliably generalize the learnt structure to natural data. Training on synthetic data may bias the model toward artifacts introduced by the synthetic constructions themselves. As a consequence, models fail to generalize to natural test suites, which are structurally and contextually more complex—where complexity refers to the presence of additional constituents that are not directly relevant to the task. In contrast, models trained on natural data show more robust cross-condition generalization, performing competitively not only on natural test suites but also on synthetic ones.

Natural sentences—here extracted from UD treebanks—exhibit greater lexical variability (e.g. proper nouns or hapaxes) and additional linguistic objects surrounding the target syntactic configuration. Rather than hindering learning, however, this variability appears to promote abstraction of the core elements of the task, although not reaching ceiling performance in the testing suites.

The Universal Dependencies (UD) framework –as a particularly curated and cross-linguistically valid example of structured and annotated data in general– plays a central methodological role in this study. By extracting instances through dependency relations and morphological features, we were able to construct datasets that are both naturally occurring and structurally controlled. Crucially UD allows the same query to be executed consistently across different languages and text genres. This makes it possible to retrieve comparable structures in a uniform way, without redefining language-specific search criteria each time. In this sense, UD and structured annotations in general function as a bridge between raw language data and the need to build theoretically motivated, structured diagnostics paradigms for natural data.

6. Related Work

Language models have demonstrated good performance on lexically restricted verb alternations and thematic roles, especially in English (Kann et al., 2019; Warstadt et al., 2019; Wilson et al., 2023; Samo et al., 2023; Proietti et al., 2022). Our work expands computational approaches to the passive alternation (Sasano et al., 2013; Leong and Linzen, 2023).

BLMs share objectives with datasets relying on synthetic or carefully controlled materials, designed to target specific grammatical phenomena under controlled conditions, often using minimal pairs or carefully crafted paradigms to probe the linguistic knowledge of large language models across languages (Warstadt et al., 2019, 2020; Xiang et al., 2021; Suijkerbuijk et al., 2025). Unlike datasets of minimal pairs, though, the BLM puzzle presents a complex structure and expands the minimal contrasts to many dimensions of variation.

While synthetic data allow broad and systematic coverage, they can introduce distributional biases, encouraging models to rely on superficial regularities rather than abstract rules (Nadăş et al., 2025; Griffiths et al., 2024). Our results partly resonate with the findings of Zhang and Pavlick (2025), who show that models trained on synthetic data may continue to rely on superficial heuristics that generalize poorly to targeted evaluations, even when overall benchmark performance improves.

Natural data drawn from annotated and structured resources provide rich linguistic information at the word (Batsuren et al., 2022) or sentence level (Nivre, 2015; De Marneffe et al., 2021), offering authentic variation and contextual richness. Such data allow researchers to systematically probe language models, designing tasks that implicitly target complex grammatical phenomena, including agreement (Jumelet et al., 2025) and argument structure alternations in morphologically-rich languages (Samo and Merlo, 2026b). By bridging naturalistic data and controlled evaluation, structured corpora provide a principled framework for investigating the inner workings of language models.

7. Conclusion

Our study shows that while synthetic data allow models to achieve near-perfect performance on in-distribution tasks, they fail to support robust generalization to natural language. In contrast, models trained on natural sentences, systematically retrieved from Universal Dependencies, capture the

underlying patterns of passive alternation and generalize effectively across both natural and synthetic test sets. These findings highlight the enduring value of authentic, linguistically-grounded data for evaluating large language models: variability and complexity in natural corpora encourage models to learn abstract structural generalizations rather than memorizing surface patterns. Structured and annotated natural data remain crucial for rigorous linguistic evaluation and for probing the true syntactic and semantic knowledge of pretrained models cross-linguistically.

Limitations

Future work could address the limitations of this contribution by expanding language coverage and alternation phenomena, exploring additional models and architectures, and performing comprehensive validation, as well a human upperbound.

Ethics

We used datasets derived from publicly available corpora, which may include content such as news articles and other publicly accessible materials. It is important to note that these datasets may contain sensitive or potentially upsetting topics. We acknowledge that such content could be distressing to some individuals. We encourage users to approach the results with awareness of these considerations.

Acknowledgements

We gratefully acknowledge the support of this work by the Swiss National Science Foundation, through grant SNF Advanced grant TMAG-1_209426 to PM.

8. Bibliographical References

Lorenzo Alfieri and Fabio Tamburini. 2016. (Almost) Automatic Conversion of the Venice Italian Treebank into the Merged Italian Dependency Treebank Format. In *Proceedings of the Third Italian Conference on Computational Linguistics (CLiC-IT 2016)*, pages 19–23, Napoli, Italy.

Aixiu An, Chunyang Jiang, Maria A. Rodriguez, Vivi Nastase, and Paola Merlo. 2023. BLM-AgrF: A New French Benchmark to Investigate Generalization of Agreement in Neural Networks. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1363–1374, Dubrovnik, Croatia.

Khuyagbaatar Batsuren, Omer Goldman, Salam Khalifa, Nizar Habash, Witold Kieraś, Gábor Bella, Brian Leonard, Garrett Nicolai, Kyle Gorman, Yustinus Ghanggo Ate, Maria Ryskina, Sabrina Mielke, Elena Budianskaya, Charbel El-Khaissi, Tiago Pimentel, Michael Gasser, William Abbott Lane, Mohit Raj, Matt Coler, Jaime Rafael Montoya Samame, Delio Siticonatzi Camaiteri, Esaú Zumaeta Rojas, Didier López Francis, Arturo Oncevay, Juan López Bautista, Gema Celeste Silva Villegas, Lucas Torroba Hennigen, Adam Ek, David Guriel, Peter Dirix, Jean-Philippe Bernardy, Andrey Scherbakov, Aziyana Bayyr-ool, Antonios Anastasopoulos, Roberto Zariquiey, Karina Sheifer, Sofya Ganieva, Hilaria Cruz, Ritván Karahóǧa, Stella Markantonatou, George Pavlidis, Matvey Plugaryov, Elena Klyachko, Ali Salehi, Candy Angulo, Jatayu Baxi, Andrew Krizhanovsky, Natalia Krizhanovskaya, Elizabeth Salesky, Clara Vania, Sardana Ivanova, Jennifer White, Rowan Hall Maudslay, Josef Valvoda, Ran Zmigrod, Paula Czarnowska, Irene Nikkarinen, Aelita Salchak, Brijesh Bhatt, Christopher Straughn, Zoey Liu, Jonathan North Washington, Yuval Pinter, Duygu Ataman, Marcin Wolinski, Totok Suhardijanto, Anna Yablonskaya, Niklas Stoehr, Hossep Dolatian, Zahroh Nuriah, Shyam Ratan, Francis M. Tyers, Edoardo M. Ponti, Grant Aiton, Aryaman Arora, Richard J. Hatcher, Ritesh Kumar, Jeremiah Young, Daria Rodionova, Anastasia Yemelina, Taras Andrushko, Igor Marchenko, Polina Mashkovtseva, Alexandra Serova, Emily Prud'hommeaux, Maria Nepomniashchaya, Fausto Giunchiglia, Eleanor Chodroff, Mans Hulden, Miikka Silfverberg, Arya D. McCarthy, David Yarowsky, Ryan Cotterell, Reut Tsarfaty, and Ekaterina Vylomova. 2022. UniMorph 4.0: Universal Morphology. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 840–855, Marseille, France. European Language Resources Association.

Cristina Bosco, Felice Dell'Orletta, Simonetta Montemagni, Manuela Sanguinetti, and Maria Simi. 2014. The Evalita 2014 Dependency Parsing Task. In *Proceedings of CLiC-it 2014 and EVALITA 2014*, pages 1–8, Pisa, Italy. Pisa University Press.

Joan Bresnan. 2016. Lifetime Achievement Award: Linguistics: The Garden and the Bush. *Computational Linguistics*, 42(4):599–617.

Dominique Brunato, Felice Dell'Orletta, and Giulia Venturi. 2022. Linguistically-Based Comparison of Different Approaches to Building Corpora for Text Simplification: A Case Study on Italian. *Frontiers in Psychology*, Volume 13 - 2022.

Luigi Burzio. 1986. *Italian Syntax: A Government and Binding Approach*. Reidel, Dordrecht.

Hao Chen, Abdul Waheed, Xiang Li, Yidong Wang, Jindong Wang, Bhiksha Raj, and Marah I. Abdin. 2024. On the Diversity of Synthetic Data and its Impact on Training Large Language Models. *arXiv preprint*, arXiv: 2410.15226.

Noam Chomsky. 1965. *Aspects of the Theory of Syntax*. MIT Press, Cambridge, MA.

Marco Civico. 2025. Linguistic statistical universals: comparing computer- and human-generated texts. *International Journal of Digital Humanities*, 7(1):1–37.

Marie-Catherine De Marneffe, Christopher D Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal dependencies. *Computational linguistics*, 47(2):255–308.

Charles J. Fillmore, Christopher R. Johnson, and Miriam R. L. Petruck. 2003. Background to FrameNet. *International Journal of Lexicography*, 16(3):235–250.

Richard Futrell, Ethan Wilcox, Takashi Morita, Peng Qian, Miguel Ballesteros, and Roger Levy. 2019. Neural Language Models as Psycholinguistic Subjects: Representations of Syntactic State. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 32–42, Minneapolis, Minnesota. Association for Computational Linguistics.

Edward Gibson and Evelina Fedorenko. 2013. The Need for Quantitative Methods in Syntax and Semantics Research. *Language and Cognitive Processes*, 28(1–2):88–124.

Anita Goyal and Samira Mahmoud. 2024. A Systematic Review of Synthetic Data Generation Techniques for NLP. *Electronics*, 13(17):3509.

Thomas L Griffiths, Jian-Qiao Zhu, Erin Grant, and R Thomas McCoy. 2024. Bayes in the Age of Intelligent Machines. *Current Directions in Psychological Science*, 33(5):283–291.

Bruno Guillaume. 2021. Graph Matching and Graph Rewriting: GREW tools for Corpus Exploration, Maintenance and Conversion. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 168–175, Online. Association for Computational Linguistics.

Bruno Guillaume, Marie-Catherine de Marneffe, and Guy Perrier. 2019. Conversion et améliorations de corpus du français annotés en universal dependencies. *Traitement Automatique des Langues*, 60(2):71–95. HAL: hal-02267418.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint*, arXiv:2501.12948.

Paul J Hopper and Sandra A Thompson. 1980. Transitivity in Grammar and Discourse. *Language*, pages 251–299.

Chunyang Jiang, Giuseppe Samo, Vivi Nastase, and Paola Merlo. 2024. BLM-It - Blackbird Language Matrices for Italian: A CALAMITA Challenge. In *Proceedings of the 10th Italian Conference on Computational Linguistics (CLiC-it 2024)*, pages 1135–1143, Pisa, Italy. CEUR Workshop Proceedings.

Martin Josifoski, Marija Sakota, Maxime Peyrard, and Robert West. 2023. Exploiting Asymmetry for Synthetic Training Data Generation: SynthIE and the Case of Information Extraction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1555–1574, Singapore. Association for Computational Linguistics.

Jaap Jumelet, Leonie Weissweiler, Joakim Nivre, and Arianna Bisazza. 2025. MultiBLiMP 1.0: A Massively Multilingual Benchmark of Linguistic Minimal Pairs. *arXiv preprint arXiv:2504.02768*.

Katharina Kann, Alex Warstadt, Adina Williams, and Samuel R. Bowman. 2019. Verb Argument Structure Alternations in Word and Sentence Embeddings. In *Proceedings of the Society for Computation in Linguistics (SCiL) 2019*, pages 287–297.

Sabine Lehmann, Stephan Oepen, Sylvie Regnier-Prost, Klaus Netter, Veronika Lux, Judith Klein, Kirsten Falkedal, Frederik Fouvry, Dominique Estival, Eva Dauphin, Herve Compagnion, Judith Baur, Lorna Balkan, and Doug Arnold. 1996. TSNLP - Test Suites for Natural Language Processing. In *COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics*.

Cara Su-Yi Leong and Tal Linzen. 2023. Language models can learn exceptions to syntactic rules. In *Proceedings of the Society for Computation in Linguistics 2023*, pages 133–144, Amherst, MA. Association for Computational Linguistics.

Beth Levin. 1993. *English Verb Classes and Alternations A Preliminary Investigation*. University of Chicago Press, Chicago and London.

Paola Merlo. 2023a. Blackbird language matrices (BLM), a new task for rule-like generalization in neural networks: Can large language models pass the test? In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8119–8152, Singapore. Association for Computational Linguistics.

Paola Merlo. 2023b. Blackbird language matrices (BLM), a new task for rule-like generalization in neural networks: Motivations and Formal Specifications. *arXiv*, arXiv: 2306.11444.

Mihai Nadăş, Laura Dioşan, and Andreea Tomescu. 2025. Synthetic data generation using large language models: Advances in text and code. *IEEE Access*, 13:134615–134633.

Vivi Nastase, Giuseppe Samo, Chunyang Jiang, and Paola Merlo. 2024a. Exploring Italian sentence embeddings properties through multitasking. In *Proceedings of the Tenth Italian Conference on Computational Linguistics (CLiC-it 2024)*, pages 620–630, Pisa, Italy. CEUR Workshop Proceedings.

Vivi Nastase, Giuseppe Samo, Chunyang Jiang, and Paola Merlo. 2024b. Exploring syntactic information in sentence embeddings through multilingual subject-verb agreement. In *Proceedings of the Tenth Italian Conference on Computational Linguistics (CLiC-it 2024)*, pages 631–643, Pisa, Italy. CEUR Workshop Proceedings.

Vivi Nastase, Giuseppe Samo, Chunyang Jiang, and Paola Merlo. 2025. Multilingual vs. Monolingual Transformer Models in Encoding Linguistic Structure and Lexical Abstraction. In *Proceedings of the Eleventh Italian Conference on Computational Linguistics (CLiC-it 2025)*, pages 826–836, Cagliari, Italy. CEUR Workshop Proceedings.

Malvina Nissim, Danilo Croce, Viviana Patti, Pierpaolo Basile, Giuseppe Attanasio, Elio Musacchio, Matteo Rinaldi, Federico Borazio, Maria Francis, Jacopo Gili, Daniel Scalena, Begoña Altuna, Ekhi Azurmendi, Valerio Basile, Luisa Bentivogli, Arianna Bisazza, Marianna Bolognesi, Dominique Brunato, Tommaso Caselli, Silvia Casola, Maria Cassese, Mauro Cettolo, Claudia Collacciani, Leonardo De Cosmo, Maria Pia Di Buono, Andrea Esuli, Julen Etxaniz, Chiara Ferrando, Alessia Fidelangeli, Simona Frenda, Achille Fusco, Marco Gaido, Andrea Galassi, Federico Galli, Luca Giordano, Mattia Goffetti, Itziar Gonzalez-Dios, Lorenzo Gregori, Giulia Grundler, Sandro Iannaccone, Chunyang Jiang, Moreno La Quatra, Francesca Lagioia, Soda Marem Lo, Marco Madeddu, Bernardo Magnini, Raffaele Manna, Fabio Mercorio, Paola Merlo, Arianna Muti, Vivi Nastase, Matteo Negri, Dario Onorati, Elena Palmieri, Sara Papi, Lucia Passaro, Giulia Pensa, Andrea Piergentili, Daniele Potertì, Giovanni Puccetti, Federico Ranaldi, Leonardo Ranaldi, Andrea Amelio Ravelli, Martina Rosola, Elena Sofia Ruzzetti, Giuseppe Samo, Andrea Santilli, Piera Santin, Gabriele Sarti, Giovanni Sartor, Beatrice Savoldi, Antonio Serino, Andrea Seveso, Lucia Siciliani, Paolo Torroni, Rossella Varvara, Andrea Zaninello, Asya Zanollo, Fabio Massimo Zanzotto, Kamyar Zeinalipour, and Andrea Zugarini. 2025. Challenging the Abilities of Large Language Models in Italian: a Community Initiative. *arXiv preprint*, arXiv: 2512.04759.

Joakim Nivre. 2015. Towards a Universal Grammar for Natural Language Processing. In *International conference on intelligent text processing and computational linguistics*, pages 3–16. Springer.

Juri Opitz, Shira Wein, and Nathan Schneider. 2025. Natural language processing relies on linguistics. *Computational Linguistics*, 51(3):1009–1032.

Mattia Proietti, Gianluca Lebani, and Alessandro Lenci. 2022. Does BERT Recognize an Agent? Modeling Dowty's Proto-Roles with Contextual Embeddings. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4101–4112, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Malka Rappaport Hovav and Beth Levin. 2024. Variable Agentivity: Polysemy or underspecification. *Glossa: a journal of general linguistics*, 9(1).

John C. Raven. 1938. Standardization of progressive matrices. *British Journal of Medical Psychology*, 19:137–150.

Tanya Reinhart. 1981. Pragmatics and linguistics: An analysis of sentence topics. *Philosophica*, 27.

Giuseppe Samo and Paola Merlo. 2021. Intervention effects in clefts: a study in quantitative computational syntax. *Glossa: a Journal of General Linguistics*, 6(1):145.

Giuseppe Samo and Paola Merlo. 2026a. Datasets for Verb Alternations across Languages: BLM Templates and Data Augmentation Strategies. *arXiv*, arXiv: 2603.15295.

Giuseppe Samo and Paola Merlo. 2026b. Modelling the morphology of verbal paradigms: A

case study in the tokenization of Turkish and Hebrew. In *Proceedings of the Second Workshop Natural Language Processing for Turkic Languages (SIGTURK 2026)*, pages 82–94, Rabat, Morocco. Association for Computational Linguistics.

Giuseppe Samo, Vivi Nastase, Chunyang Jiang, and Paola Merlo. 2023. BLM-s/lE: A structured dataset of English spray-load verb alternations for testing generalization in LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12276–12287, Singapore. Association for Computational Linguistics.

Ryohei Sasano, Daisuke Kawahara, Sadao Kurohashi, and Manabu Okumura. 2013. Automatic knowledge acquisition for case alternation between the passive and active voices in Japanese. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1213–1223, Seattle, Washington, USA. Association for Computational Linguistics.

Job Schepens, Hanna Woloszyn, Nicole Marx, and Benjamin Gagl. 2025. Can Large Language Models Generate Useful Linguistic Corpora?: A Case Study of the Word Frequency Effect in Young German Readers. *Open Mind*, 9:1597–1656.

Carson T Schütze, Jon Sprouse, Robert J Podesva, and Devyani Sharma. 2013. Judgment data. *Research methods in linguistics*, pages 27–50.

Djamé Seddah and Marie Candito. 2016. Hard time parsing questions: Building a questionbank for french. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia.

Anna Siewierska. 2013. Passive constructions (v2020.4). In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Zenodo.

Jon Sprouse and Diogo Almeida. 2017. Design sensitivity and statistical power in acceptability judgment experiments. *Glossa: a journal of general linguistics*, 2(1):14.

Michelle Suijkerbuijk, Zoë Prins, Marianne de Heer Kloots, Willem Zuidema, and Stefan L. Frank. 2025. BLiMP-NL: A Corpus of Dutch Minimal Pairs and Acceptability Judgments for Language Model Evaluation. *Computational Linguistics*, 51(4):1267–1301.

Francesca Volpato. 2010. *The Acquisition of Relative Clauses and phi-Features: Evidence from Hearing and Hearing-impaired Populations*. Ph.D. thesis, Università Ca'Foscari Venezia.

Alex Warstadt, Yu Cao, Ioana Grosu, Wei Peng, Hagen Blix, Yining Nie, Anna Alsop, Shikha Bordia, Haokun Liu, Alicia Parrish, Sheng-Fu Wang, Jason Phang, Anhad Mohananey, Phu Mon Htut, Paloma Jeretic, and Samuel R. Bowman. 2019. Investigating BERT's Knowledge of Language: Five Analysis Methods with NPIs. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2877–2887, Hong Kong, China. Association for Computational Linguistics.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do Llamas Work in English? On the Latent Language of Multilingual Transformers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15366–15394, Bangkok, Thailand. Association for Computational Linguistics.

Michael Wilson, Jackson Petty, and Robert Frank. 2023. How abstract is linguistic generalization in large language models? experiments with argument structure. *Transactions of the Association for Computational Linguistics*, 11:1377–1395.

Beilei Xiang, Changbing Yang, Yu Li, Alex Warstadt, and Katharina Kann. 2021. CLiMP: A benchmark for Chinese language model evaluation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2784–2790, Online. Association for Computational Linguistics.

Lingze Zhang and Ellie Pavlick. 2025. Does Training on Synthetic Data Make Models Less Robust? *arXiv preprint*, arXiv: 2502.07164.

Appendix

#Args	Voice	ST	Pattern	Without
2	Act	Q	V -[nsubj]-> Ag; V-[obj]-> Pat; Q [form="?"]	Y [upos=VERB]
1	Act	Decl	V -[nsubj]-> Ag; V-[obj]-> Pat	Q [form="?"]; Y [upos="VERB"]
2	Act	Q	V -[nsubj]-> Ag; Q [form="?"]	V-[obj]-> Pat; Y [upos="VERB"]
1	Act	Decl	V -[nsubj]-> Ag	V-[obj]-> Pat; Q [form="?"]; Y [upos="VERB"]
2	Pass	Q	V -[nsubj:pass]-> Pat; V-[obl:agent]-> Ag; Q [form="?"]	Y [upos="VERB"]
1	Pass	Decl	V -[nsubj:pass]-> Pat; V-[obl:agent]-> Ag	Q [form="?"]; Y [upos="VERB"]
1	Pass	Q	V -[nsubj:pass]-> Pat; Q [form="?"]	V-[obl:agent]-> Ag; Y [upos="VERB"]
1	Pass	Decl	V -[nsubj:pass]-> Pat	V-[obl:agent]-> Ag; Q [form="?"]; Y [upos="VERB"]

Table 1: Queries to retrieve natural instatiation example from UD treebanks with Grew-Match (Guillaume, 2021). #Args = number of arguments, ST = Sentence Type, Act = Active Voice, Pass = Passive Voice, Q = question, D = declarative.

Subevent Structure as a Predictor of Entity Identity Change in Procedural Text

Kyeongmin Rim, James Pustejovsky
Brandeis University
Waltham, MA, USA
krim@brandeis.edu, jamesp@brandeis.edu

Abstract

We test whether the subevent structure encoded in VerbNet-GL predicts entity identity change in procedural text, using only the verb's lexical specification and no training data. From each VN verb class's SEMANTICS block we extract an aspectual classification and an I/O count, yielding a predicted *dynamic event topology* (DET). On the observation side, ten large language models (LLMs) annotate per-entity *dynamic object mode* (DOM) labels over ∼3100 OpenPI steps, from which we derive observed DET for comparison. The VN-only predictor achieves 67.4% precision (F1 = 0.35) for transformation, showing that formal subevent structure carries genuine predictive signal only for the most common topology, but low performance for other topologies. For events VN predicts as having no result state, only 24% are confirmed as no-change by silver, indicating that the remaining outcomes arise from the argument side of the composition. These results provide empirical evidence that event semantics is distributed across predicate and argument: the VN supplies the subeventual skeleton, but is not sufficient to determine the final outcome.

Keywords: VerbNet, entity state tracking, event semantics, procedural text, distributed compositionality, co-composition

1. Introduction

A central claim of Generative Lexicon (GL) theory (Pustejovsky, 1995) is that semantic composition is not a static process of function application, but a generative one involving *co-composition*. Under this mechanism, the verb's event interpretation is sensitive to the *qualia structure* of its arguments, a structured representation encoding how an object comes into being (AGENTIVE), what it is for (TELIC), its formal type (FORMAL), and its constitutive parts (CONSTITUTIVE). In the classic example of *bake*, the event resolves to transformation for a *potato* (a natural kind lacking an AGENTIVE quale) but shifts to emergence for a *cake* (an artifact whose AGENTIVE quale licenses a creation reading). To test this linguistic theory, procedural text provides an ideal test case, as it describes the *lifecycle of entities* through sequences of actions that transform their state (e.g., flour → dough → bread), and datasets such as OpenPI (Tandon et al., 2020) provide per-step state annotations that record these empirical outcomes.

VerbNet (VN; Kipper et al., 2008) organizes ∼300 verb classes by shared syntactic and semantic behavior. Its extension with GL subevent semantics (hereafter VN-GL; Brown et al., 2019) decomposes each class into temporally ordered subevents ($e_1, e_2, \ldots$), uniquely equipping it to serve as a formal basis for this analysis. In the GL framework, qualia roles act as a representational overlay on this subevent structure, specifying the predicative constraints and oppositions associated with each phase. This subeventual skeleton is designed to capture precisely the kind of entity change that OpenPI annotates: each event can be modeled as an input/output (I/O) transformation (Rim et al., 2023) where participants enter a process and exit in a potentially altered state, even when the result is not textually mentioned (Ježek and Melloni, 2011).

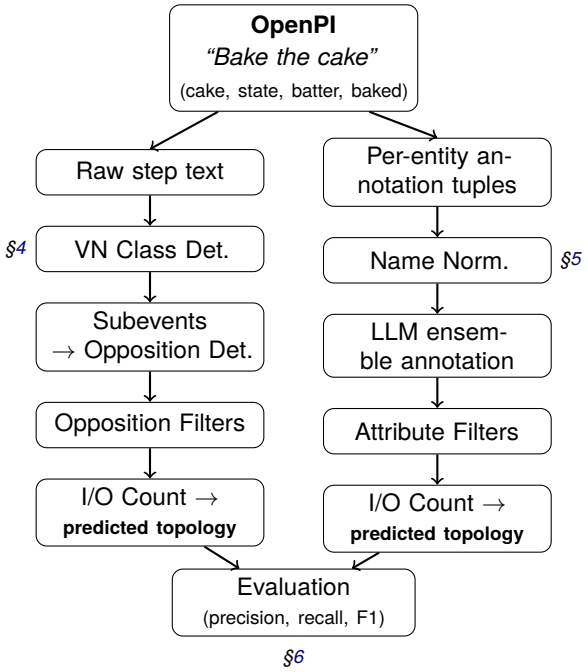


Figure 1: Evaluation Design: Two independent paths derive step-level DET from the same OpenPI data, compared at alignment.

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 52–63

While VN-GL subevent structure defines the structural constraints of such transformations, we argue that the verb's lexical specification remains inherently underspecified. Thus, the final interpretation, including the identity of the resulting state, must emerge from a mutual specification between the verb and its participants. This suggests that event meaning is effectively *distributed* across the predicate and its arguments, rather than being concentrated within the verb alone.

In this paper, we present the first empirical evaluation of this distributed interpretation by testing the predictive reach of the verb's formal contribution (its subevent structure) in isolation. We ask: how much of the observed entity change in procedural text is predicted by the verb alone? This verb-only baseline constitutes a controlled ablation of argument information, directly measuring the "compositional gap" that co-composition is intended to fill.

The answer matters because systematic failures in a verb-only model reveal exactly where argument qualia is required, providing empirical evidence for GL's distributed interpretation. Moreover, a symbolic resource offers falsifiable, inspectable predictions and a finite scope for targeted extension.

We define a step-level *dynamic event topology* (DET) that characterizes the structural shape of entity identity change at each procedural step, and derive it via two independent paths: (1) *predicted* DET, extracted symbolically from VN-GL subevent structure (aspectual classification and I/O count), requiring no training data; and (2) *observed* DET, derived from per-entity *dynamic object mode* (DOM) labels that ten LLMs assign using OpenPI's crowdsourced attribute tuples, with no VN-GL information in the prompt. Comparing the two measures how much of the observed entity change the verb's lexical specification alone can account for. Our results show that while verb-level semantics reliably predicts transformation in transition events, it systematically fails in three cases: low emergence precision, unmodeled state changes in process events, and argument-qualia overrides in destruction-class verbs. These are precisely the points where GL predicts that argument-driven co-composition is required to determine the event outcome, providing empirical evidence that event semantics is distributionally composed across predicate and argument structure.

Our contributions are:

1. A symbolic method for deriving dynamic event topologies from VN-GL subevent structure (aspectual classification + I/O count), requiring no training data.
2. A silver-standard dynamic object mode (DOM) dataset over OpenPI, produced by majority vote across ten LLMs and assessed against manual gold annotation.
3. An empirical evaluation of VN-GL's predictive reach, with error categories grounded in GL's co-composition theory.

2. Related Work

2.1. Entity State Tracking in Procedural Text

Entity state tracking focuses on monitoring how participants change through sequences of actions. ProPara (Dalvi et al., 2018) introduced the task with a closed vocabulary (location and existence) over scientific process paragraphs. The OpenPI dataset (Tandon et al., 2020) scaled it to open-vocabulary state-change tuples ($\sim$30000) over WikiHow procedures, where each tuple records an entity, an attribute, and its before/after values (e.g., *onion*, *state*, *whole*, *diced*). Quality issues in OpenPI's original crowdsourcing have been addressed by filtering unreliable state changes (Wu et al., 2023) and canonicalizing entity and attribute strings (Zhang et al., 2024).

Neural entity trackers have progressed from structured conditional random field (CRF) models (Gupta and Durrett, 2019b) through transformer-based approaches (Gupta and Durrett, 2019a) to interactive entity networks (Tang et al., 2020). On the knowledge-based side, ProComp (Clark et al., 2018) and Lexis (Kazeminejad et al., 2021) predict state changes symbolically from VerbNet rules without training data, while PROPOLIS (Faghihi et al., 2023) integrates symbolic semantic parsing with neural components. More recently, EvEntS ReaLM (Spiliopoulou et al., 2022) tackles entity state reasoning via language models. Our work differs from all of these in that we do not predict entity states directly; instead, we test whether a formal lexical resource can predict the event's change topology (DET) from verb semantics alone.

2.2. Formal Models of Subevent Structure

Event structure theories decompose verbs into temporal phases. Vendler's (1967) four-way aspectual classification (state, activity, accomplishment, achievement) was enriched by Moens and Steedman (1988) with transitions and culmination points, providing a direct predecessor to Pustejovsky's subevent decomposition. Pustejovsky (2013) formalizes this dynamic event model, which was then implemented in VerbNet (Kipper et al., 2008), built on Levin's (1993) diathesis-based verb classification, by Brown et al. (2018), later refined in Brown et al. (2019) and Brown et al. (2022). The resulting VN-GL framework utilizes subevent variables and predicate opposition to model phase transitions.

Tu et al. (2024) adapt this opposition structure to build graph representations augmenting Abstract Meaning Representation (AMR) with subevent annotations. Fine-grained intra-class semantic distinctions have been documented by Kazeminejad et al. (2022).

Our work relies on GL's theory of co-composition (Pustejovsky, 1995), which holds that arguments actively contribute to an event's interpretation. We adopt the Dynamic Argument Structure (DAS) taxonomy (Jezek, 2017) and the Generalized Result Role thesis (Ježek and Melloni, 2011) to map verb-level subevents to the entity-level changes observed in procedural data. Crucially, resources like VN-GL localize this structure in the verb class, while we argue that event semantics is distributionally composed across both predicate and arguments.

2.3. Silver Data Adjudication and Distillation

Generating high-quality annotations for open-vocabulary state tracking has proven difficult via crowdsourcing (Tandon et al., 2020; Wu et al., 2023). Recent work has shifted toward using Large Language Models (LLMs) as high-fidelity meta-annotators, even outperforming human crowd workers in structured semantic tasks (Gilardi et al., 2023). Following the Models-Vote Prompting paradigm (Oniani et al., 2023), ensemble methods are increasingly used to mitigate family-specific hallucinations and biases (Kamen and Kamen, 2025). Our work builds on (Rim and Pustejovsky, 2026), which establishes that the relationship between state outcomes and semantic structures is systematic and recoverable. However, while typical silver-labeling pipelines stop at model consensus, we introduce a layer of *theory-informed post-processing* (Veselovsky et al., 2023): GL's distinction between extrinsic change (location, possession) and intrinsic identity change overrides model consensus for entities whose annotations involve only spatial or ownership attributes.

3. Task Definition

3.1. Dynamic Object Mode

On the observation side, each OpenPI entity, represented by its (entity, attribute, before, after) tuples, receives a label that we call the *dynamic object mode* (DOM): the mode of identity change that the entity undergoes at a given step. The term reflects the argument side of the distributed composition: the verb (predicate) determines the structural shape of the event (DET), while the DOM records what happens to each participant. Step-level DET is then derived mechanically from these per-entity DOM labels (Section 5.4), enabling comparison against the VN-GL predicted DET.

We ground the DOM taxonomy in the Dynamic Argument Structure (DAS) framework of Jezek (2017), which characterizes five modes for event participants: *initiation* (brought into existence), *termination* (ceases to exist), *modification* (an attribute value changes), *transfer* (information moves through a medium), and *persistence* (present but unaffected). We narrow DAS *modification* to type-level change ($T_1 \rightarrow T_2$): an entity is **transformed** only if it becomes a different kind of thing (e.g. batter $\rightarrow$ cake). Attribute-level changes (location, temperature) that preserve the entity's type fall under **persistent**. The DOM taxonomy is:

- **Created**: the entity comes into being at this step.
- **Terminated**: the entity ceases to exist at this step.
- **Transformed**: the entity persists with identity preserved but undergoes a type-level change ($T_1 \rightarrow T_2$).
- **Persistent**: the entity is present and participated but not meaningfully affected.

A fifth label, **bystander**, flags entities whose OpenPI tuples are too noisy for meaningful classification (e.g., coreference errors, abstract concepts, annotator artifacts); these are excluded from downstream DET derivation and all evaluation metrics.

3.2. Dynamic Event Topology

Where DOM describes what happens to each entity, the *dynamic event topology* (DET) describes the structural shape of the event as a whole, the predicate side of the distributed composition.

DET is grounded in two theoretical foundations in Generative Lexicon (GL). First, GL's aspectual classification (Pustejovsky, 1995) groups Vendler's (1967) four classes into three: *states* (static, no change), *processes* (activities, no result state), and *transitions* (accomplishments and achievements, which encode a result state). Only transitions receive a DET assignment; since stative verbs rarely appear as main predicates in procedural text, only the transition/process distinction is operative here. Second, for transitions, DET is derived from the predicative opposition that GL's FORMAL quale encodes: by counting which participant roles appear across the opposition pairs in the subevent structure, we obtain the number of entities entering (n_{in}) versus exiting (n_{out}) the result state. This yields five topologies:

- **Transformation** ($1 \rightarrow 1$): one entity enters and one exits with altered type.

- **Emergence** ($0 \rightarrow n$): no input entity; one or more new ones appear in the result state.
- **Dissolution** ($n \rightarrow 0$): one or more entities enter but none persists in the result.
- **Convergence** ($n \rightarrow 1,\ n \geq 2$): multiple inputs merge into a single output.
- **Divergence** ($1 \rightarrow n,\ n \geq 2$): a single input divides into multiple outputs.

Process and state events receive no DET assignment.

3.3. Evaluation Design

The core evaluation compares *predicted* DET, derived symbolically from VN-GL subevent structure (Section 4.2), against *observed* DET, derived mechanically from per-entity DOM labels annotated by an LLM ensemble over OpenPI steps (Section 5.4). The two derivations are fully independent: the prediction side uses no OpenPI annotations, and the observation side uses no VN-GL information (Figure 1). Their alignment measures how much of the observed entity change the verb's lexical specification alone can account for.

4. Predicted DET from VN-GL

4.1. Motivation for a Resource-Based Approach

An obvious alternative is to skip VN-GL entirely and ask an LLM to predict entity state changes directly. This would likely achieve higher accuracy. But this work is not a state-tracking benchmark; it is a test of a linguistic formalism.

A formal resource provides what an LLM cannot. First, *falsifiability*: VN-GL makes a specific, inspectable claim per verb class; when a class mispredicts, the failure traces to a specific formal structure (a missing result subevent, an unmodeled co-patient), not to opaque model weights. Second, *controlled ablation*: the verb-only DET baseline is an intentional removal of qualia information from the argument side, directly testing GL's distributed co-compositionality thesis: how much event meaning resides in the predicate versus the argument? An LLM conflates both, having seen verb-argument combinations in training, and cannot isolate the predicate's contribution. Third, *resource auditing*: VN-GL is a finite, curated resource ($\sim$300 classes, $\sim$6000 lemmas); understanding its predictive reach provides a concrete roadmap for targeted extension.

4.2. DET Prediction Pipeline

Given a sentence from OpenPI dataset, the prediction pipeline derives a DET of the events in the sentence through three stages. No OpenPI entity or state annotation is used; the prediction is entirely verb-determined.

Stage 1: VerbNet class assignment The VerbNet Parser (VNP; Gung and Palmer, 2021) maps the step's main verb to a VerbNet class and extracts surface thematic roles from the syntactic frame. From the VN 3.4 XML for the assigned class, we select the frame whose declared syntax roles best match VNP's surface roles (ties favor the earlier frame, as VN orders frames from canonical to specialized).

Stage 2: Opposition pair detection and aspectual gating From the selected frame's SEMANTICS block, each `PRED` element is extracted with its subevent variable, polarity (`bool="!"` for negated), thematic role arguments, and any `PredSpecific` arguments. VN-GL uses three event variable types (Brown et al., 2022): `E` (overall event variable, holds throughout the entire event); and two temporally ordered subevent types: $\ddot{e}_n$ (process subevents, unbounded activity) and e_n (state-transition subevents). We consider only predicates at plain e_n variables that have at least one ThemRole argument, skipping `E` and $\ddot{e}$ predicates.

These predicates are grouped by (predicate name, ThemRole). Within each group, we look for *opposition pairs*: the same predicate on the same role at two different subevent indices, with evidence of opposition. Two patterns are recognized:[1]

- **Pattern 1** (negation): opposite polarity at different indices (e.g., $\neg$`cooked(`e_1`, Patient)` and `cooked(`e_4`, Patient)`);
- **Pattern 2** (PredSpecific): different `PredSpecific` values at different indices (e.g., `has_val(`e_1`, Patient, Initial_State)` and `has_val(`e_3`, Patient, Result)`).

For each pair, the entry at the earlier index is the *initial side*; the later is the *result side*.

If at least one opposition pair is found, the event is a *transition*; otherwise it is a *process* and receives no DET assignment. This gatekeeper is deliberately conservative: the entity-change rate

[1] Across all 1602 frames in VN 3.4, these two patterns account for all detected opposition structure (1225 negation-based pairs across 37 predicates; 21 PredSpecific-based pairs across 3 predicates). 692 additional cases of the same predicate and role at different indices with same polarity and no PredSpecific difference are continuity tracking, not oppositions. Two predicates (`created`, `exist`) appear in the VN 3.4 XML but are absent from the published predicate inventory of Brown et al. (2022); the published inventory lists `degraded` where the XML uses `degradation_material_integrity`.

observed in process events (Section 6.2) measures the compositional gap that argument qualia must fill.

Stage 3: I/O role counting and existence classification Not all detected opposition pairs constitute type-level identity change. In GL, location change is extrinsic (Pustejovsky, 1995): it alters a relation between the entity and an external domain, not the entity's intrinsic type; possession transfer similarly changes ownership without altering the object itself. Pairs on these predicates encode spatial or ownership changes that preserve entity identity. These pairs still contribute to the aspectual gate (they confirm the event has subevent structure), but their roles are excluded from I/O counting.[2]

From the remaining identity-relevant pairs, we count distinct roles: n_{in} from initial-side entries whose role is in the input set (Patient, Theme, Co-Patient, Co-Theme, Material, Asset), and n_{out} from result-side entries in the output set (Patient, Theme, Result, Product). When the class declares a Co-Patient or Co-Theme in its THEMROLES, n_{in} is raised to at least 2.

Raw counting treats every role as both input and output if it appears on both sides of a pair. However, some oppositions describe an entity coming into or going out of existence, rather than persisting through a state change. For such roles, counting them on both sides is incorrect: an entity that ceases to exist is input-only (should not count toward n_{out}), and an entity that comes into existence is output-only (should not count toward n_{in}).

To identify these cases, we manually tagged each of the 163 VN 3.4 predicates (Brown et al., 2022) with its *existence semantics*: *existence* predicates (8) whose positive form entails the entity is present, *cessation* predicates (4) whose positive form entails it is no longer present, and all others with no existence implication. [3]

For each negation-based pair (Pattern 1), the predicate's tag and polarity at each side determine the per-role adjustment:

- entity ceases to exist (e.g., `alive` → ¬`alive`, or ¬`destroyed` → `destroyed`): the role is removed from n_{out};
- entity comes into existence (e.g., ¬`created` → `created`): the role is removed from n_{in};
- all other Pattern 1 pairs and all Pattern 2 pairs: no adjustment.

The raw and adjusted $(n_{\text{in}}, n_{\text{out}})$ determines the predicted DET from the topology definitions in Section 3.2. Namely, we evaluate the existence semantics rule as an ablation in Section 6.2.

For example, the SEMANTICS block for COOKING-45.3 contains ¬`cooked(`e_1`, Patient)` and `cooked(`e_4`, Patient)`. These form a negation-based opposition pair on Patient. Since `cooked` has no existence implication, no adjustment is applied: Patient contributes to both n_{in} and n_{out}, yielding $(1, 1) \rightarrow$ transformation.

5. Observed DET from Silver DOM

5.1. OpenPI Overview

OpenPI (Tandon et al., 2020) provides ∼30000 open-vocabulary state-change tuples over 810 WikiHow documents. Each tuple records an entity, an attribute, and its before/after values at a given procedural step (e.g., *onion*, *state*, *whole*, *diced*). The dataset covers six WikiHow categories: Food & Entertaining, Home & Garden, Hobbies & Crafts, Cars & Other Vehicles, Sports & Fitness, and a residual category.

Raw OpenPI tuples record attribute-level state changes (e.g., temperature, location, shape) but do not directly encode DOM, which abstracts over these attributes to characterize the entity's biographical mode of identity change (*created*, *terminated*, *transformed*, or *persistent*). The same entity may appear with multiple attribute tuples at a single step (e.g., *onion*: state whole→diced, location cutting-board→pan), and surface entity strings are unnormalized ("the deep fryer" and "deep fryer" are separate entries). Quality issues in the original crowdsourcing have been documented by Wu et al. (2023), who found ∼32% of state changes unreliable, and by Zhang et al. (2024), who attempted entity canonicalization with limited success.

5.2. Silver DOM Annotation

We derive silver-quality DOM labels via LLM ensemble annotation, with post-processing informed by GL's distinction between extrinsic and intrinsic change.

Annotation distillation For each step, models receive the document goal, step text, and the set of raw (attribute, before, after) tuples for each entity.

[2] This exclusion is symmetric with the observation side, where OpenPI entities whose attribute tuples consist solely of location or possession changes are relabeled to **persistent** (Section 5.2).

[3] Location: `has_location`, `has_orientation`, `has_position`. Possession: `has_possession`, `has_information`. Existence (positive = entity present): `alive`, `appear`, `be`, `created`, `exist`, `give_birth`, `procreate`, `visible`. Cessation (positive = entity gone): `destroyed`, `disappear`, `suffocated`, `voided`. Classification based on predicate definitions in Appendix A and semantic clusters in Appendix C of Brown et al. (2022).

	Annotator	Docs	Steps	Entities	cre	ter	tra	per	bys
	Majority vote	**639**	**3101**	**14642**	**4.8**	**2.1**	**37.3**	**40.7**	**15.1**
Silver (per-model)	Gemma-2B	*(same as above)*		8682	55.1	4.5	23.3	15.5	1.7
	Gemma-7B			14423	8.0	3.5	66.7	11.9	9.9
	Gemma-3-12B			14596	5.7	3.8	56.0	23.1	11.3
	Qwen2.5-14B			14543	10.3	2.5	69.0	4.0	14.2
	Qwen3-14B			14588	6.2	3.5	74.4	11.2	4.6
	Haiku-4.5			14638	4.9	7.9	28.3	38.7	20.2
	Sonnet-4.6			14621	4.0	6.4	18.5	47.3	23.8
	GPT-4.1-mini			14606	8.2	5.3	24.6	40.7	21.2
	GPT-5-mini			14611	3.0	3.5	17.9	59.5	16.1
	GPT-5.2			14614	3.7	2.2	14.6	51.3	28.1
Gold		20	104	532	2.4	1.5	56.2	21.2	18.6

Table 1: DOM label distribution (%) for silver (per-model and majority vote) and gold annotations, after post-processing. Entity counts vary across models because each model may fail to parse or label a subset of entities. Column headers: cre = created, ter = terminated, tra = transformed, per = persistent, bys = bystander.

They are instructed to adjudicate these noisy, low-level strings into the DOM taxonomy (Section 3.1); the full prompt and a few-shot example are given in Appendix A.[4] Crucially, no VN-GL information or subevent definitions are included in the prompt; the LLMs classify entity change solely from the OpenPI tuples, ensuring that the observation side is fully independent of the prediction side. This treats the LLM as a meta-annotator, distilling clean, high-level semantic features from crowdsourced strings (Gilardi et al., 2023).

Ensemble adjudication Following the Models-Vote Prompting paradigm (Oniani et al., 2023), we employ ten instruction-tuned LLMs: five local GPU models (Gemma-2B, Gemma-7B, Gemma-3-12B (Gemma Team et al., 2024), Qwen2.5-14B, Qwen3-14B (Qwen Team, 2025)) and five commercial APIs (GPT-4.1-mini, GPT-5-mini, GPT-5.2, Claude-Haiku-4.5, Claude-Sonnet-4.6). This "wisdom of the crowd" mitigates family-specific hallucinations and biases inherent in single-model pipelines (Kamen and Kamen, 2025). The final silver label for each entity is determined by plurality vote across the ensemble; ties are broken by a fixed priority ordering that favors more informative modes (transformed $>$ created $>$ terminated $>$ persistent).

Pre-processing: entity normalization OpenPI entity mentions are free-form text annotations with no standardized form. We apply minimal normalization (lowercasing, stripping articles, collapsing whitespace), which merges "the deep fryer" with "deep fryer" but not "deep fryer" with "fryer." Zhang et al. (2024) attempted LLM-based entity canonicalization via GPT-3.5-turbo with 3-shot prompting, but achieved an entity clustering F1 of only 0.495, meaning roughly half of coreference links were missed. Given this limited reliability, we opt for the simpler deterministic normalization and treat residual duplicates as separate entities.

Post-processing: non-identity-change override As discussed in Section 4.2, location and possession changes do not constitute type-level identity change. On the observation side, we force-relabel any entity whose OpenPI annotations consist solely of location-related attributes (`location`, `orientation`, `position`, `placement`, `movement`, `motion`, `direction`, `distance`) or possession-related attributes (`ownership`, `possession`) to **persistent**, regardless of model consensus. This affects $\sim$21% of entities (3112 of 14642).

Silver dataset statistics The silver dataset comprises 639 documents (all training-split documents with at least one entity tuple), 3101 steps, and 14642 entities. Table 1 shows the per-model and majority-vote label distributions. Two model clusters emerge: transformation-heavy models (Qwen, Gemma-7B/12B; 56–74%) versus conservative models (Claude, GPT; 15–28%), with the majority vote mediating at 37%/41% (transformed/persistent). The non-identity-change override (Section 5.2) relabels 3112 entities (21.3%) to persistent.

5.3. Human Verification

5.3.1. Concreteness Filtering

Not all OpenPI procedures describe physical transformations. To focus human verification on documents where entity identity change is expected, we

[4]The prompt uses simplified label names (e.g., "transform" for **transformed**, "destroyed" for **terminated**) that were established before the final taxonomy was formalized; we verified that models interpret these consistently across all ten LLM annotators.

implement a two-layer concreteness filter. Layer 1 uses lexico-syntactic heuristics: a keyword blocklist for video game titles, spaCy (Honnibal et al., 2023) named entity recognition (NER) for WORK_OF_ART entities, and exclusion of goals whose root verb is abstract (*be*, *seem*, *involve*). Layer 2 computes mean noun concreteness via the Brysbaert et al. (2014) norms ($\sim$40k lemmas, 1–5 scale), normalized to $[0, 1]$ and thresholded at > 0.8. This identifies 422 of 639 documents as concrete. The filter is also used as a post-hoc robustness check on the full evaluation (Section 6.2).

5.3.2. Gold Annotation

We randomly sampled 20 documents from the 422 concrete-filtered set; however, four (20%) were discarded and redrawn after manual inspection revealed non-physical procedures that passed the automatic filter (e.g., clothing selection, exercise routines), indicating that the concreteness heuristic is a coarse screen rather than a reliable classifier. All steps within each document were annotated, yielding 104 steps and 532 entities (Table 1, bottom row).

Annotation guidelines include a decision flowchart, edge-case policies, and worked examples; the full text will be released with the dataset. Two graduate students independently labeled the sample using the five-way DOM taxonomy. Inter-annotator agreement was moderate (Cohen's $\kappa = 0.41$; $\kappa = 0.48$ excluding bystander), largely depressed by noise in the underlying OpenPI tuples, as many borderline entities receive inconsistent attribute annotations in the source data, making DOM assignment genuinely ambiguous. The two sets were adjudicated into a single gold standard used for all evaluations in Section 6.

5.4. Observed DET Derivation

Step-level DET is derived mechanically from per-entity DOM counts; it is not a separate annotation. If any entity is **transformed**, or exactly one is **terminated** and one **created**, the step receives **transformation**. **Created**-only steps yield **emergence**; **terminated**-only steps yield **dissolution**. When multiple entities are **terminated** and at least one is **created**, the step is **convergence**; the reverse yields **divergence**. Steps where all entities are **persistent** receive no DET label.

6. Evaluation

6.1. Silver Label Quality

Table 3 compares each model's DOM labels against the adjudicated gold standard (532 entities, **bystander** excluded). The models fall into two clusters. Transformation-heavy models (Qwen3-14B, Qwen2.5-14B, Gemma-7B) reach 0.63–0.68 entity-level accuracy with strong **transformed** F1 (0.74–0.81) but near-zero **persistent** F1. Conservative models (Claude, GPT) balance **transformed** and **persistent** F1 (0.48–0.65 and 0.50–0.56) at lower overall accuracy. The majority vote mediates these clusters, achieving the highest step-level DET accuracy (0.654) with balanced per-label F1, consistent with the complementary-bias findings of Oniani et al. (2023).

Table 2 confirms this cluster structure via pairwise Cohen's κ at the entity level. Intra-cluster agreement is moderate to substantial (transformation-heavy: κ = 0.48–0.60; conservative: κ = 0.55–0.63), while inter-cluster agreement is lower (κ = 0.21–0.44). Gemma-2B is an outlier with uniformly low agreement (κ = 0.16–0.30). This diversity supports the ensemble design.

6.2. DET Prediction Evaluation

Table 4 compares verb-only DET predictions (Section 4.2) against the silver DET derived from majority-vote DOM labels. Of 2940 overlapping steps, 863 (29.4%) agree on the DET label.

Among transition events, **transformation** $(1{\to}1)$ achieves 67.4% precision over 596 steps: when VN-GL predicts a single-participant transition, the silver observation agrees in roughly two out of three cases, without any training data or argument information.[5] The overall accuracy (29.4%) falls below a majority-class baseline that always predicts transformation (58.8%). This deficit is driven by two factors: process events ($n = 1655$), which receive no DET assignment and only 24% of which are confirmed as no-change by silver, and convergence over-prediction (388 steps at 0% precision) from Co-role declarations in THEMROLES. However, where the predictor commits to transformation, it outperforms the baseline (67.4% vs. 58.8%), and the non-transformation predictions provide structurally informative error categories that a majority-class baseline cannot.

The process-event gap has two sources. First, VNP may misassign a process class to a verb that VN-GL actually encodes as a transition (a parser error that could be addressed with improved class disambiguation). Second, some verbs are genuinely aspectually unbounded but still produce observable state changes on their arguments in procedural context (*stir the batter* changes the batter's state despite *stir* being a process verb); this is precisely

[5] Restricting to the 422 concrete-filtered documents (Section 5.3.1) yields 70.0% transformation precision over 404 steps (27.5% overall, 1946 steps), suggesting that non-physical procedures introduce modest noise but do not materially affect the findings.

	Model	1	2	3	4	5	6	7	8	9	10
1.	Gemma-2B	—									
2.	Gemma-7B	.30	—								
3.	Gemma-3-12B	.25	.48	—							
4.	Qwen2.5-14B	.28	.51	.51	—						
5.	Qwen3-14B	.28	.56	.55	.60	—					
6.	Haiku-4.5	.22	.33	.44	.39	.36	—				
7.	Sonnet-4.6	.17	.25	.36	.30	.26	.63	—			
8.	GPT-4.1-mini	.23	.29	.40	.36	.32	.62	.57	—		
9.	GPT-5-mini	.16	.21	.32	.26	.23	.55	.57	.58	—	
10.	GPT-5.2	.16	.20	.30	.27	.23	.56	.61	.57	.61	—

Table 2: Pairwise entity-level Cohen's κ across the ten silver annotator models. Mean κ = 0.38.

Model	N	κ	DOM	DET
Gemma-2B	241	.026	.261	.183
Gemma-7B	377	.122	.647	.538
Gemma-3-12B	408	.105	.598	.510
Qwen2.5-14B	396	.088	.634	.538
Qwen3-14B	423	.190	.683	.587
Haiku-4.5	400	.275	.570	.500
Sonnet-4.6	387	.180	.478	.433
GPT-4.1-mini	400	.190	.505	.433
GPT-5-mini	415	.171	.489	.500
GPT-5.2	382	.164	.474	.452
Majority	**417**	**.219**	**.585**	**.654**

Table 3: Individual model and majority-vote DOM labels evaluated against adjudicated gold (bystander excluded). N = entities with labels from model and gold overlap; κ = Cohen's kappa; DOM = entity-level accuracy; DET = step-level accuracy.

the case where GL predicts that the argument's qualia must supply the result interpretation that the verb alone does not encode.

6.3. Discussion

6.3.1. Error Sources

VerbNet Parser accuracy All DET predictions depend on VNP (Gung and Palmer, 2021) correctly assigning a VerbNet class. VNP was trained on declarative sentences, but OpenPI procedures are imperative (e.g., *Dice the onion*). We use the Stanza dependency parser (Qi et al., 2020) to detect imperative sentences and prepend "you" before parsing (*you dice the onion*), converting imperative to pseudo-declarative form. VNP accuracy on this augmented input was not measured; misassigned classes propagate directly into wrong DET predictions.

OpenPI noise Crowdsourced tuples contain ∼32% unreliable annotations (Wu et al., 2023), surface entity strings are unnormalized, and coreference is unresolved. This noise affects both gold inter-annotator agreement (IAA; $\kappa = 0.41$) and the silver quality ceiling, as models must adjudicate noisy tuples into clean DOM labels. The two behavioral clusters in Table 3 reflect genuine ambiguity in the source data as much as model disagreement.

Cardinality extraction Convergence overprediction is the largest single error source: 388 steps predicted as convergence ($n_{\text{in}} \geq 2$), none matching silver. This arises from the Co-role adjustment: classes declaring Co-Patient or Co-Theme in THEMROLES receive $n_{\text{in}} \geq 2$, but in procedural text these classes rarely produce actual convergence events. Dissolution predictions (28 steps, 32.1% precision) are recovered by the existence classification (Section 4.2), but silver observes transformation for 11/28 of these steps, indicating that argument qualia overrides the predicate's encoded termination in procedural context. Divergence (73 steps at 0% precision) is a further error source where VN-GL structural cardinality does not match the observed topology.

6.3.2. Distributed Compositionality

The transformation precision (67.4% over 596 steps) shows that the verb's subevent structure carries substantial predictive information for transition events, without access to any argument information. Three systematic failure modes indicate where the predicate alone is insufficient.

VN-GL predicts emergence for 175 steps, with 51 matching silver (29.1% precision), while silver observes 468 emergence steps overall. Emergence in procedural text is typically implicit (*flour + water → dough*), where the result entity's existence depends on the argument's qualia structure, specifically the AGENTIVE quale that licenses a creation reading (Pustejovsky, 1995). The existence classification detects predicates like `created` and `exist` and removes the role from n_{in}, but this captures only a fraction of actual emergence events.

Only 24% of steps classified as process events are confirmed as no-change by silver. Verbs such as *stir*, *spread*, and *rub* are aspectually unbounded but produce observable state changes on their arguments in procedural context. The verb's aspectual class yields no DET prediction; the argument's participation must force a result interpretation.

Of 28 dissolution predictions, 9 match silver (32.1% precision). These are cases where VN-GL encodes entity termination via cessation pred-

	Raw				+ Exist. class.			
DET	**N**	**P**	**R**	**F1**	**N**	**P**	**R**	**F1**
transformation	684	.658	.260	.373	596	.674	.233	.346
emergence	115	.348	.085	.137	175	.291	.109	.159
dissolution	0	–	–	–	28	.321	.043	.077
convergence	388	.000	.000	.000	388	.000	.000	.000
divergence	73	.000	.000	.000	73	.000	.000	.000
no DET	1680	.239	.762	.364	1680	.239	.762	.364
acc.		**.303**				**.294**		
baseline		*.588*				*.588*		

Table 4: DET prediction alignment with ablation. *Raw*: opposition pairs detected, I/O roles counted directly. + *Exist. class.*: existence/cessation tags applied to adjust cardinality for emergence and dissolution. "no DET" includes both process events ($n = 1655$) and transitions with $(0, 0)$ cardinality ($n = 25$). Baseline: always predict transformation.

icates (`destroyed`, `suffocated`), and the existence classification correctly identifies the role as input-only. Where dissolution is mispredicted, silver typically observes transformation, indicating that argument qualia overrides the predicate's encoded termination in procedural context.

Together, these failure modes delineate the boundary between what the predicate contributes and what the argument must supply, consistent with GL's claim that event meaning is distributed across predicate and argument structure.

6.4. Future Work

The cardinality-based DET heuristic could be replaced with a neural classifier trained over SEMANTICS blocks, predicting DET directly from predicate structure rather than relying on role-counting rules. On the argument side, integrating noun-level semantic resources (e.g., WordNet hypernym chains, qualia structure annotations (Pustejovsky et al., 2009)) could extend the symbolic pipeline to capture the argument qualia that verb-only prediction misses, testing whether VN-GL combined with a noun resource improves DET prediction without training data. A neuro-symbolic encoder for GL qualia structure could learn dense representations of predicate-argument interactions, combining the interpretability of formal resources with the coverage of distributional models.

7. Conclusion

We evaluated VerbNet-GL subevent structure as a symbolic predictor of entity identity change in procedural text. From VN-GL SEMANTICS blocks, we derived aspectual classifications and I/O counts to produce symbolic DET predictions over ∼3100 OpenPI steps, evaluated against silver DOM labels from a ten-model LLM ensemble. The verb-only predictor achieves 67.4% precision for transformation over 596 steps (29.4% overall), indicating that formal subevent structure carries predictive signal for transition events. Systematic failures—low emergence precision, process-event state changes, and argument-qualia overrides in destruction-class verbs—correspond to cases where GL theory predicts that argument qualia is required to complete the event interpretation, providing evidence that event semantics is distributed across predicate and argument structure. The VN-DET code, gold and silver datasets, and annotation guidelines will be released upon publication.

8. Limitations

Our evaluation depends on the VerbNet Parser for class assignment, which was trained on declarative sentences and has not been evaluated on imperative procedural text; the pseudo-declarative conversion (prepending "you") is a heuristic mitigation whose accuracy remains unmeasured, and misassigned classes propagate directly into DET prediction errors. The silver standard, while validated against gold annotation, inherits noise from OpenPI's crowdsourced tuples (∼32% unreliable per Wu et al. (2023)). Among the 2077 mispredictions, convergence over-prediction (388 steps from Co-role declarations), divergence over-prediction (73 steps), and process events (only 24% of 1655 steps confirmed as no-change by silver) are the most prominent residual error sources. Finally, the evaluation covers only WikiHow procedures, which skew toward physical tasks; generalization to other procedural domains (e.g., scientific protocols with different verb distributions, or assembly manuals relying more on nominal predicates) remains untested.

9. Bibliographical References

Susan Windisch Brown, Julia Bonn, James Gung, Annie Zaenen, James Pustejovsky, and Martha Palmer. 2019. VerbNet Representations: Subevent Semantics for Transfer Verbs. In *Proceedings of the First International Workshop on Designing Meaning Representations*, pages 154–163, Florence, Italy. Association for Computational Linguistics.

Susan Windisch Brown, Julia Bonn, Ghazaleh Kazeminejad, Annie Zaenen, James Pustejovsky, and Martha Palmer. 2022. Semantic Representations for NLP Using VerbNet and the Generative Lexicon. *Frontiers in Artificial Intelligence*, 5.

Susan Windisch Brown, James Pustejovsky, Annie Zaenen, and Martha Palmer. 2018. Integrating Generative Lexicon Event Structures into VerbNet. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. 2014. Concreteness ratings for 40 thousand generally known English word lemmas. *Behavior Research Methods*, 46(3):904–911.

Peter Clark, Bhavana Dalvi, and Niket Tandon. 2018. What Happened? Leveraging VerbNet to Predict the Effects of Actions in Procedural Text.

Bhavana Dalvi, Lifu Huang, Niket Tandon, Wentau Yih, and Peter Clark. 2018. Tracking State Changes in Procedural Text: A Challenge Dataset and Models for Process Paragraph Comprehension. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1595–1604, New Orleans, Louisiana. Association for Computational Linguistics.

Hossein Rajaby Faghihi, Parisa Kordjamshidi, Choh Man Teng, and James Allen. 2023. The Role of Semantic Parsing in Understanding Procedural Text.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. 2023. ChatGPT outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30):e2305016120.

James Gung and Martha Palmer. 2021. Predicate Representations and Polysemy in VerbNet Semantic Parsing. In *Proceedings of the 14th International Conference on Computational Semantics (IWCS)*, pages 51–62, Groningen, The Netherlands (online). Association for Computational Linguistics.

Aditya Gupta and Greg Durrett. 2019a. Effective use of transformer networks for entity tracking. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Aditya Gupta and Greg Durrett. 2019b. Tracking discrete and continuous entity state for process understanding. In *Proceedings of the Third Workshop on Structured Prediction for NLP*. Association for Computational Linguistics.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, Adriane Boyd, and Henning Peters. 2023. spaCy: Industrial-strength Natural Language Processing in Python. Zenodo.

Elisabetta Jezek. 2017. Dynamic Argument Structure. *Linguistic Issues in Language Technology*, 15(3).

Elisabetta Ježek and Chiara Melloni. 2011. Nominals, polysemy, and co-predication. *Journal of Cognitive Science*, 12(1):1–31.

Ariel Kamen and Yakov Kamen. 2025. Majority rules: LLM ensemble is a winning approach for content categorization. *arXiv preprint arXiv:2511.15714*.

Ghazaleh Kazeminejad, Martha Palmer, Susan Windisch Brown, and James Pustejovsky. 2022. Componential analysis of English verbs. volume 5. Frontiers.

Ghazaleh Kazeminejad, Martha Palmer, Tao Li, and Vivek Srikumar. 2021. Automatic Entity State Annotation using the VerbNet Semantic Parser. In *Proceedings of the Joint 15th Linguistic Annotation Workshop (LAW) and 3rd Designing Meaning Representations (DMR) Workshop*, pages 123–132, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Karin Kipper, Anna Korhonen, Neville Ryant, and Martha Palmer. 2008. A large-scale classification of English verbs. *Language Resources and Evaluation*, 42(1):21–40.

Beth Levin. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press.

Marc Moens and Mark Steedman. 1988. Temporal ontology and temporal reference. *Computational Linguistics*, 14(2):15–28.

David Oniani, Jordan Hilsman, Hang Dong, Shiven Verma, Fengyi Gao, and Yanshan Wang. 2023. Large language models vote: Prompting for rare disease identification. *arXiv preprint arXiv:2308.12890*.

James Pustejovsky. 1995. *The Generative Lexicon*. MIT Press.

James Pustejovsky. 2013. Dynamic Event Structure and Habitat Theory. In *Proceedings of the 6th International Conference on Generative Approaches to the Lexicon (GL2013)*, pages 1–10, Pisa, Italy. Association for Computational Linguistics.

James Pustejovsky, Jessica Moszkowicz, Olga Batiukova, and Anna Rumshisky. 2009. GLML: Annotating Argument Selection and Coercion. In *Proceedings of the Eight International Conference on Computational Semantics*, pages 169–180, Tilburg, The Netherlands. Association for Computational Linguistics.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108. Association for Computational Linguistics.

Qwen Team. 2025. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Kyeongmin Rim and James Pustejovsky. 2026. Missing links: LLM-augmentation of event triggers of state changes in the OpenPI dataset. In *Proceedings of the 2026 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2026)*. European Language Resources Association.

Kyeongmin Rim, Jingxuan Tu, Bingyang Ye, Marc Verhagen, Eben Holderness, and James Pustejovsky. 2023. The Coreference under Transformation Labeling Dataset: Entity Tracking in Procedural Texts Using Event Models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12448–12460, Toronto, Canada. Association for Computational Linguistics.

Evangelia Spiliopoulou, Artidoro Pagnoni, Yonatan Bisk, and Eduard Hovy. 2022. EvEntS ReaLM: Event Reasoning of Entity States via Language Models.

Niket Tandon, Keisuke Sakaguchi, Bhavana Dalvi, Dheeraj Rajagopal, Peter Clark, Michal Guerquin, Kyle Richardson, and Eduard Hovy. 2020. A Dataset for Tracking Entities in Open Domain Procedural Text. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6408–6417, Online. Association for Computational Linguistics.

Jizhi Tang, Yanqiang Qu, and Yansong Li. 2020. Understanding procedural text using interactive entity networks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Jingxuan Tu, Timothy Obiso, Bingyang Ye, Kyeongmin Rim, Keer Xu, Liulu Yue, Susan Windisch Brown, Martha Palmer, and James Pustejovsky. 2024. GLAMR: Augmenting AMR with GL-VerbNet Event Structure. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7746–7759, Torino, Italy. ELRA and ICCL.

Zeno Vendler. 1967. Verbs and times. In *Linguistics in Philosophy*, pages 97–121. Cornell University Press.

Veniamin Veselovsky, Manoel Horta Ribeiro, Akhil Arora, Martin Josifoski, Ashton Anderson, and Robert West. 2023. Generating faithful synthetic data with large language models: A case study in computational social science. *arXiv preprint arXiv:2305.15041*.

Xueqing Wu, Sha Li, and Heng Ji. 2023. OpenPI-C: A Better Benchmark and Stronger Baseline for Open-Vocabulary State Tracking. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7213–7222, Toronto, Canada. Association for Computational Linguistics.

Li Zhang, Hainiu Xu, Yue Yang, Shuyan Zhou, Weiqiu You, Manni Arora, and Chris Callison-Burch. 2024. OpenPI2.0: An improved dataset for entity tracking in texts. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 415–431. Association for Computational Linguistics.

A. Silver Annotation Prompt

The following prompt and few-shot examples are used for all ten LLMs in the silver annotation ensemble.

System Prompt

```
You classify entity identity
changes in procedural text.

Given a procedural goal, the
steps completed so far, the cur-
rent step, and a list of entities
with their observed attribute
changes, classify each entity's
identity status at this step.

Labels (use exactly one per en-
tity):

- transform: the entity's TYPE
changes while its identity is
preserved. You would call it by
a different name afterward. Ex-
amples: raw dough becomes baked
bread, whole onion becomes diced
onion, empty bowl becomes bowl
of batter. Moving an entity to
a different location or changing
its temperature alone is NOT a
transform.

- created: the entity comes into
existence at this step; it did
not exist as a distinct object
before.

- destroyed: the entity ceases
to exist at this step; it is used
up, discarded, or absorbed into
something else.

- unchanged: the entity is a
physical object that partici-
pates in the step but is not
meaningfully altered. It is the
same kind of thing before and
after. Tools being used (knife,
oven, bowl), entities that merely
change location, and entities
with superficial attribute dif-
ferences (e.g. same temperature)
are unchanged.

- irrelevant: the entity is not
actually involved in this step,
or is not a physical object. Use
for: agents/users performing
the procedure (e.g. "you", "your
hands"), abstract concepts (e.g.
"knowledge", "confidence"), and
entities whose attribute change
is nonsensical given the step.

For each entity listed, judge its
status at the current step. Re-
spond with exactly one line per
entity in the format:

entity_name: label

Do not add explanation or commen-
tary.
```

Few-shot example (1 of 5)

```
User:
Goal: Make Mango Marinade
Procedure so far: Prepare and
measure all ingredients. / Add
all the ingredients to a food
processor.
Current step: Blend thoroughly
until you achieve a smooth tex-
ture.
Entities:
ingredients: texture=lumpy-
>smooth, texture=intact->blended
food processor: state=on-
>through blending
mixture: texture=chunky-
>smoother

Assistant:
ingredients: transform
food processor: unchanged
mixture: transform
```

The complete set of five few-shot examples is available in the released codebase.

Fast and Flexible Example-Based Treebank Search with Vector Symbolic Architectures

Adam Roussel
Department of Linguistics
Ruhr University Bochum
Bochum, Germany
adam.roussel@rub.de

Abstract

In this paper we introduce an example-based method for exploring dependency treebanks that is based on principles of vector symbolic architectures. It leverages key properties of this framework to provide fast and flexible search capabilities, since all combinations of query parameters can be compared with a given parse tree in parallel via a single vector operation. The framework also allows for graded similarity and the natural integration of various kinds of information, such as word embeddings. After some background on the framework and an explanation of our implementation, we provide a few examples of the system's output and draw comparisons to similar applications.

Keywords: hyperdimensional computing, vector symbolic architectures, treebanks, example-based search

1. Introduction

Treebanks contain a great wealth of data that can be very helpful in the development of linguistic theories. Making effective use of that information can be challenging, in part due to the size of a typical treebank, but largely due to complex structure of the data itself. Thus such datasets are usually queried using specialized query languages that enable one to specify very precisely what structures are sought. This precision is not always an asset. Sometimes it's not clear exactly what constraints are necessary, and, for some phenomena, there is a natural range of variation that can be difficult to specify precisely. In such cases, a robust and flexible approach to treebank exploration can be useful.

In our contribution, we describe an approach to searching dependency treebanks that is example-based and leverages properties inherent to the framework of vector symbolic architectures to make this process both fast and flexible. Our approach is lightweight and can easily be run locally on ordinary computer hardware, and every query, regardless of its apparent complexity, takes the same amount of time. This amount of time is strictly linear with respect to the number of sentences in the treebank. Our approach is also robust and flexible. Rather than users needing to specify upfront which constraints may be relaxed in order to improve recall, our VSA search approach is fuzzy, since all possible combinations of features are queried simultaneously. The system works by measuring the similarity between a query and all of the sentences in the treebank, on which basis it then provides a ranking of all the sentences. The more properties that overlap between the query example and the candidate, the higher its similarity, and thus its ranking, will be.

Beyond these immediate practical advantages, this framework has intriguing potential in other respects. In principle, many different kinds of data can be represented, such that results can be preferred when they are similar to the query with respect to more than just their syntax. For instance, numbers can be represented as similar when they are similar in quantity, and words can be made more similar on the basis of their overall distribution, in both cases introducing semantic factors to influence the search results.

We will begin, in Section 2, with a comparison of our approach with similar efforts from the literature, before offering a brief introduction to vector symbolic architectures in general (Section 3). Then, we describe the specific encoding strategy that we use in our approach in Section 4. Section 5 shows the results for some example queries, as well as some discussion and analysis of these results. Finally, in Section 6, we will discuss some of our plans for extending and improving our approach in future work. All of the software described in this paper is available from https://codeberg.org/aroussel/vsatreebank under a free software license (GPLv3).

2. Related Work

In existing tools for searching dependency treebanks, such as Gärtner et al. (2013) and Luotolahti et al. (2015), queries can be understood as a set of constraints. A candidate is only considered a match if all of the constraints hold. This is often exactly what one wants from a search tool. In other scenarios, one may wish for a query to be somewhat more flexible, or one may not know exactly in what order the constraints should be loosened

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 64–73

in order to retrieve the desired results. A further complication is that search tools, especially those for searching treebanks, often require the use of specialized query languages.

Attempting to address these issues, example-based search tools have been introduced, such as the Linguist's Search Engine (Resnik and Elkiss, 2005) and GrETEL (Augustinus et al., 2012; Odijk et al., 2017), which are more closely related to the strategy we are presenting here. In this style of search tool, the user begins with an example of the construction of interest, which is parsed automatically. The user then has the option to delete uninteresting parts of the parse tree or to stipulate that the lemma must match or that only the POS must match, etc. In this way, constraints are manually relaxed, so that similar but non-identical matches can be found.

In contrast, the approach that we are proposing does not necessarily require any such adjustment of the query, since the comparisons are fuzzy by default. All components of a parse tree are compared simultaneously. Any non-matching elements register as mere noise, and matching elements increase the similarity between the query and a candidate. A nice side-effect of this fuzzy matching is that it provides a workaround for sentences that may have been mis-parsed by the automatic parser, in that such errors won't necessarily lead to relevant candidate sentences being missed. These properties follow naturally from the properties of VSA/HDC representations.

There are further advantages that VSA vector representations have to offer. Karlgren (2023) argue that they offer the promise of using linguistically meaningful and structured representations in contexts that ordinarily demand vectors. Accordingly this application to searching treebanks draws its strengths from these two aspects. Because complex data structures are representable with VSAs, they can be used to represent parse trees, and because vector representations allow for efficient, graded comparisons, those parse trees can be quickly ranked according to their relevance.

Widdows and Cohen (2015) apply techniques of VSAs in order to construct a knowledge base of relational triples, which can be explored through the use of specially constructed query vectors, an approach they call "predication-based semantic indexing" or PSI. Specifically, the authors describe the application of PSI to a biomedical knowledge base, so the predication triples describe relations like "insulin TREATS diabetes mellitus". One can construct vectors that represent questions, like "what treats diabetes?", and then receive in the list of the nearest neighbors some possible answers. Such query vectors can also be constructed on the basis of analogies, as in Kanerva's classic "what is the dollar of Mexico?" example (Kanerva, 2009).

This general approach of constructing a query vector and finding the nearest neighbors in a database is the same basic operating principle of our treebank search approach, as described in this paper. We provide a more detailed description of this procedure in Section 4 below.

3. Background on vector symbolic architectures

Vector symbolic architectures, also known as hyperdimensional computing (HDC), are a computing framework that rely on certain properties of high-dimensional vector spaces, typically 10 000 or more. (For a more detailed introduction, see: Kanerva (2009); Schlegel et al. (2022); Kleyko et al. (2022).) In such high-dimensional spaces, with say 10 000 dimensions, there are exactly 10 000 orthogonal vectors, but there are very many more nearly orthogonal vectors, such that any two randomly chosen vectors are very likely nearly orthogonal. From this it follows that we can readily generate a new vector for any given symbol just by choosing a random vector from this high-dimensional space, and each of these symbolic vectors is distinguishable from the others in that the similarity between them, according to an appropriately chosen metric, is ≈ 0, that is, they are nearly orthogonal. By contrast, correlated vectors, according to context, may correspond to equivalence, similarity, or set membership.

In addition to a vector space, vector symbolic architectures also define a few key operations in this space.[1] The first, termed "bundling" and represented by $+$, takes two vectors and yields a vector that is similar to both of the input vectors. It is also known as "superposition". Then there is a "binding" operation, represented by $\otimes$, that takes two vectors and yields a vector that is dissimilar to both of them. The inverse of the binding operation, "unbinding", is represented by $\oslash$. In logical terms, bundling can be thought of as akin to $\vee$ and binding akin to $\wedge$. There is also an additional operation, permutation, represented as ρ^n, indicating the application of this operation n times. Permutation is a unary operation that produces a vector that is dissimilar to the input vector. With these basic building blocks, various data structures can be implemented, such as key–value sets, sequences, or graphs.

There are various ways that such a vector symbolic architecture can be implemented. In this work, we use a variant that uses dense binary vectors, referred to as binary spatter code or BSC (Kanerva, 1996). In this variant, binding is implemented by

[1] In the following, we will adopt the notational conventions used in Schlegel et al. (2022).

the XOR operation, bundling is element-wise majority rule (i.e., an element is 1 if the sum across n inputs is $> n/2$), and the similarity metric is derived from the Hamming distance. In BSC, the bundling and binding operations are commutative and associative, and binding distributes over bundling. The binding operation is reversible and each vector is its own inverse in BSC, meaning that it is in theory always possible to analyse a vector and see what internal structure it has. I.e., one can unbind the key from a key–value pair to retrieve the value: $B \approx A \oslash (A \otimes B)$.

4. How the system works

4.1. Encoding syntactic dependencies

Each token is encoded as a bundle of key–value pairs. Of the standard UD fields we use `form`, `lemma`, `upos`, and `xpos`, as well as any features included in `feats`. Such a bundle of key–value pairs is thus constructed as follows:

$$v_{\text{form}} \otimes v_{\text{worm}} + v_{\text{upos}} \otimes v_{\text{NOUN}} \dots \quad (1)$$

In order to distinguish between keys and values, we permute the key vector. This is motivated by two things: One is that this acts as a kind of "namespace" for keys vs. values. By permuting v_{form}, we can keep the use of this vector as a key distinct from its use as a value. So when a token's form happens to be "form", this is represented as $\rho^1 v_{\text{form}} \otimes v_{\text{form}}$. This becomes especially relevant in BSC, since in BSC, each vector is its own inverse. The same vector, e.g. for "form", bound to itself will cancel itself out ($v_{\text{form}} \otimes v_{\text{form}} \approx \vec{0}$). The risk is perhaps minor, but to avoid such problems without necessarily needing to store an extra vector for each potential key, we simply permute each vector when it is used as a key.

$$\rho^1 v_{\text{form}} \otimes v_{\text{worm}} + \dots \quad (2)$$

4.2. Encoding strategies for dependency relations

The dependency relations are handled specially. One can think of them either as predicates that apply to entire token bundles, as in (3), in which case the equation corresponds fairly directly to the parse tree. Alternatively, due to distributivity of the operations involved, one can also think of each sequence of dependency relations as encoding a path of keys to a value, as in (4). The degree of permutation applied to each key is significant, and it reflects the order in which a sequence of edges must be followed to arrive at a particular terminal node.

$$\rho^2 v_{\text{nsubj}} \otimes (\rho^1 v_{\text{form}} \otimes v_{\text{worm}} + \rho^1 v_{\text{upos}} \otimes v_{\text{NOUN}} \dots) \quad (3)$$

$$\begin{aligned} &\rho^2 v_{\text{nsubj}} \otimes \rho^1 v_{\text{form}} \otimes v_{\text{worm}} + \\ &\rho^2 v_{\text{nsubj}} \otimes \rho^1 v_{\text{upos}} \otimes v_{\text{NOUN}} + \\ &\dots \end{aligned} \quad (4)$$

Due to the inherent properties of the binding operation, each such path of keys is distinct: A token at the path $\rho^3 v_{\text{nsubj}} \otimes \rho^2 v_{\text{det}}$ isn't similar to one found at $\rho^2 v_{\text{det}}$.

Subtree encoding. In order to facilitate the matching of properties regardless of their depth in the parse tree, the Subtree encoding strategy includes all of the subtrees for each token in the top-level bundle. That is, for an instance of *the* in a subject noun phrase, we would add the feature bundle for this token at all of the following paths:

$$\begin{aligned} &\rho^3 v_{\text{nsubj}} \otimes \rho^2 v_{\text{det}} \otimes \rho^1 v_{\text{form}} \otimes v_{\text{the}} + \\ &\rho^2 v_{\text{det}} \otimes \rho^1 v_{\text{form}} \otimes v_{\text{the}} + \\ &\rho^1 v_{\text{form}} \otimes v_{\text{the}} \dots \end{aligned} \quad (5)$$

Each token is represented by a bundle of such feature paths, corresponding to its specific location and depth in the parse tree. The representation for the whole parse tree is then the bundle of all of the token representations. The following equations give the whole representation (9) for the sentence *The worm reads*, assuming that the tokens only have word form and UPOS annotations:

$$R = (\rho^1 v_{\text{form}} \otimes v_{\text{reads}} + \rho^1 v_{\text{upos}} \otimes v_{\text{VERB}}) \quad (6)$$

$$W = (\rho^1 v_{\text{form}} \otimes v_{\text{worm}} + \rho^1 v_{\text{upos}} \otimes v_{\text{NOUN}}) \quad (7)$$

$$T = (\rho^1 v_{\text{form}} \otimes v_{\text{the}} + \rho^1 v_{\text{upos}} \otimes v_{\text{DET}}) \quad (8)$$

$$\begin{aligned} &R + \rho^2 v_{\text{nsubj}} \otimes W + W + \\ &\rho^3 v_{\text{nsubj}} \otimes \rho^2 v_{\text{det}} \otimes T + \rho^2 v_{\text{det}} \otimes T + T \end{aligned} \quad (9)$$

One potential drawback of this strategy is that it tends to lead to more components being bundled into each sentence. Thus (9) contains 6 components at the top-level. For more complex sentences with many lexical features, the number of components can often exceed 100.

Thin encoding. The Thin encoding strategy, by contrast, doesn't take any special steps to facilitate matching independently of depth. It has the advantage of requiring fewer components in total, only

3 in the case of *The worm reads* instead of 6, as with the Subtree strategy. This could mean faster encoding and less noise in the sentence vectors due to bundling. However, there is the potential disadvantage that a query will only be similar to a sentence if the phrases in the query are embedded at the same depth as in the sentence.

$$R + \rho^2 v_{\text{nsubj}} \otimes W + \rho^3 v_{\text{nsubj}} \otimes \rho^2 v_{\text{det}} \otimes T \quad (10)$$

Flat encoding. The Flat encoding strategy is an alternative way of making queries match more flexibly which requires fewer components than the Subtree strategy. In (11), only 4 components are required. According to this strategy, more deeply embedded elements are expressed by permutation alone, rather than all elements of a path being bound together into a single key. The effect is that, in contrast to the Subtree strategy, a `nsubj` relation two steps away could match, even when the intervening edges don't match, which may or may not be a desirable kind of flexibility.

$$R + \rho^2 v_{\text{nsubj}} \otimes W + \rho^2 v_{\text{det}} \otimes T + \rho^3 v_{\text{nsubj}} \otimes T \quad (11)$$

4.3. Parsing user queries and performing the search

The user provides a query in the form of an example phrase or sentence. The provided query is sent to be processed by the Spacy package (Honnibal et al., 2020). At this point it is important that a parser model is used that provides parses similar to those in the treebank to be searched. In the tests described below, we make use of treebanks annotated according to the Universal Dependencies (UD) standard (de Marneffe et al., 2021), thus it is important that Spacy models used were trained on UD data also.

Once the user query has been parsed, it is then encoded in the same manner as the sentences in the treebank. We then have one vector for the query and one for each sentence in the treebank. To perform the search, the query vector is simply compared to all of the sentence vectors, and the sentences are then ranked on the basis of the resulting similarity values.

This is the scenario that requires the least intervention on the part of the user. Thanks to the automatic parsing, the example sentence can be used as-is. On the other hand, it is also possible to construct the parse tree directly, ensuring that it has precisely the desired structure. This manually constructed parse tree can then be encoded and applied in the same fashion as otherwise.

5. Results

5.1. Resource requirements

The strengths of this approach lie in its simplicity and the minimal computing resources it requires. Yet, despite these properties, it is capable of producing the kind of results that otherwise require some computationally fairly complex indexing and memoization strategies.

The current implementation comes in at just about 400 lines of Python code. All of the tests described below were run on an ordinary, though somewhat aged, consumer-grade laptop from 2016. Most relevantly, since all of the computations described run single-threaded on the CPU, this laptop has an Intel Core i5-6267U with a base frequency of 2.9 GHz. Thus all of the timings given below can be considered fairly conservative, since most users are likely to have newer hardware than this.

The encoding process usually takes a few minutes, depending on the encoding strategy employed. The ca. 65 000 sentences of the Lassy Small corpus (van Noord et al., 2012) are encoded on the test hardware in about 8 min with the Thin strategy, 14 min with the Subtree strategy, and 17 min with the Flat strategy. This is a step that generally only needs to be done once, after which the encoded corpus vectors can be stored and retrieved from disk to be queried freely.

Such a corpus of ca. 65 000 sentences requires, when using bitpacked binary vectors with 10 000 dimensions (thus 1 250 B for each vector), about 200 MB on disk. This includes all of the vectors for each feature key, vocabulary item, etc., that is, each "symbol" in the VSA, as well as all of the vectors for each sentence in the corpus.

As mentioned above, searching the treebank involves a single vector operation for each sentence in the treebank. Both the query and each sentence are represented by a vector, and the vector operation in question measures the similarity between them. In this implementation, based on BSC, this operation is essentially the Hamming distance between the vectors, which itself amounts to XOR, more or less. Since this operation is always the same, each query takes the same amount of time, regardless of its apparent complexity. On the test hardware, a search in the Lassy Small corpus takes about 0.11 s. The amount of time it takes to run such a search is exactly linear with respect to the number of sentences in the treebank.

5.2. Example 1: NP with PP

In order to give an impression of the sort of results that one can get according to the approach we're proposing, we would like to present a comparison with an existing example-based search sys-

tem, GrETEL. For the comparison we will use this example from Augustinus et al. (2012):

(12) Het doden van olifanten is verboden.
'Killing elephants is prohibited.'

If the whole sentence is used to query the treebank as-is, then we already see that the nearest neighbors for this query are quite reasonable matches (Figure 1). They all reflect the basic syntactic structure of the query sentence, and we can also observe that many of these are not only alike in terms of their syntactic relations but also with respect to various morphological properties, such as the noun's being a deverbal noun and the prepositional object being plural. Further down the list of nearest neighbors (not included in the figure), we find sentences that contain the same basic structure but contain a number of other elements or have the words in a different order, illustrating the flexibility of the system:

(13) Voor mannen is het dragen van een korte broek in het openbaar verboden.
'For men, wearing shorts in public is prohibited.'
(WR-P-E-H-0000000049.p.37.s.4)

This sentence is the 10th best match for (12), with a similarity of 0.37.

By contrast, the GrETEL system returns 4 exact matches for a similar level of user interaction, that is, providing only the whole example sentence and accepting the default settings, which require the presence of all nodes in the query. But this is not how GrETEL is meant to be used, and Augustinus et al. (2012) show how an example sentence can be reduced to just the relevant phrase in order to turn up more relevant cases from the treebank. For the reduced query example from that paper, focusing on just the noun phrase *het doden van olifanten*, GrETEL returns 48 matches using the default settings.

The top 5 results from our system for this same phrase are shown in Figure 2. With these results, we can make a few additional observations about the properties of our approach. One is that treebank entries that only consist of matching phrases are preferred. Since there are no additional "noise" elements, the similarity is greater. Secondly, we see that some of the matches have multiple noun phrases that are similar to the query. For such entries the similarity is also greater, since there are more matching structures in the same sentence.

In both of these sets of results, we can see another way that the application we describe here differs from a conventional search engine, which is that our approach results in a ranking of all of the candidate sentences according to all of the available information. In a conventional search approach, there is no ranking, and each instance either matches or it doesn't. Thus while some of the 48 GrETEL matches are among the top 10 from our system (cf. Section 5.4 below), the highly ranked instances can be similar in unexpected and serendipitous ways, e.g. by sharing morphological features that one would not want to use as constraints.

With our VSA-based approach we can include all of the features: form, lemma, POS, number, case, etc., and all of these can be used to find matches that are as similar as possible without ruling anything out. Yet, due to the way they are combined in the high-dimensional vector representations, namely in superposition, all of the combinations of features are compared simultaneously and no decisions are required from users a priori regarding which constraints may or may not be loosened.

5.3. Example 2: Shell nouns

As further example, we consider a hypothetical case in which one is looking for examples of shell nouns (Schmid, 2000). Shell nouns are a class of nouns that serve to encapsulate possibly complex propositional content in order to make it easier to talk about. Shell nouns can typically be found in sentences like *De hoofdzaak is dat je beter wordt* 'the main thing is that you get better', i.e., there is often an abstract noun with a clausal complement.

The top 5 most similar sentences when querying the encoded Lassy Small corpus using this sentence are shown in Figure 3, in this case using the Flat encoding strategy. All 5 of these sentences contain good examples of shell nouns, and of the top 25 nearest neighbors, 17 contain shell nouns.

With the GrETEL search tool, there are no matches for *De hoofdzaak is dat je beter wordt* in Lassy Small when the whole example is submitted. But, using the tool as intended and deleting certain non-essential nodes, namely *je* and *beter* in this case, we then get 127 matches after approximately 2 s to 3 s. These matches appear to generally be good examples of shell nouns: A random sample of 20 all contained clear examples of shell nouns, with nouns such as *feit* 'fact', *voorwaarde* 'condition', or *probleem* 'problem'.

5.4. Quantitative comparison

In order to get a clearer idea of how well this approach works and how it compares to existing methods, we carried out a direct comparison between the exact matches returned by GrETEL and the ranking given by the system we propose here. Specifically, we want to know, for those cases where exact matches are returned by GrETEL for a

0.40 Het nalaten van het identificeren van risicogroepen kan tot ongewenste gezondheidsrisico's leiden.
'The cessation in identification of at-risk groups can lead to unwanted health risks.'
0.39 Het houden van gedomesticeerde dieren heet veeteelt.
'The keeping of domesticated animals is called husbandry.'
0.39 Het stellen van confronterende vragen leidt tot spanningen.
'The posing of confrontational questions leads to tensions.'
0.39 Het erkennen van fouten door de NS-directie is geen alledaags verschijnsel.
'The acknowledgment of errors by the NS management is no everyday occurrence.'
0.38 Het creëren van een winwinsituatie is in ieder geval fundamenteel.
'The creation of a win-win situation is always fundamental.'

Figure 1: Top 5 matches for the example in (12) from the Lassy Small corpus, using the Subtree encoding strategy.

0.48 Het wegnemen van organen
'The removal of organs'
0.43 Het ploegen van het land met behulp van paarden
'The plowing of the land with the use of horses'
0.42 Verzamelen van handelsstatistieken
'Collection of trade statistics'
0.42 Instellen van medicijnen
'Introduction of medications'
0.41 Toedienen van medicijnen
'Administration of medications'

Figure 2: Top 5 matches for the noun phrase *het doden van olifanten* from (12), using the Subtree encoding strategy.

given query, where those sentences appear in the overall ranking that is produced by our approach.

There are a few caveats to be aware of for this comparison. Each example sentence or phrase that is used as a query contains more features than just those that one is necessarily interested in, and these extra features can increase the ranking of some sentences. Furthermore, sentences that contain matching features are considered more similar if those matching features occur more than once, since each time an encoded feature is included in a sentence vector, its signal is amplified and made more prominent. Another challenge for this comparison are the different formats used by the two systems. Whereas GrETEL searches the constituency parses of the Lassy treebank, our system uses the UD conversion of the treebank. While the two should be generally quite close, it could nevertheless be the case that sentences that exactly match the query based on the constituency parse don't match the dependency parse given by Spacy.

While we therefore would not expect that these should match up exactly, we would expect that sentences that do contain the phenomenon of interest, as reflected in its status as an exact match, should at least be ranked higher in general than sentences that do not.

In order to quantify where the system places the exact matches, we use percentile rank. This is for two main reasons. The first is that the similarity values from each encoding strategy aren't directly comparable, since the overall degree of similarity that each encoding strategy results in differs. For the shell noun query, the average similarity for all sentences in the treebank is 0.07 for the Subtree strategy, 0.03 for Flat, and 0.02 for Thin.[2] Since the Subtree encoding strategy results in more components being bundled into each sentence vector, the similarity values tend to be higher, whereas with the Thin strategy there are fewer components to match and lower similarity values. Second, using percentile rank instead of ranks directly makes the numbers easier to interpret, since values vary from 0 to 100, with values close to 100 indicating a high ranking for a match. The results of the comparison are summarized in Table 1.

5.5. Discussion

Comparing encoding strategies. In general, as is apparent in Table 1 and Figure 5, there appears to be a stronger correlation for the shell noun query, in that sentences that match this query exactly tend to be ranked higher by our approach, regardless of the encoding strategy used. For this query, the Flat strategy has a few more outliers than the other two, but the distribution looks more or less the same for all three.

[2] Cf. the overall distribution of similarity values in Figure 6.

0.17 De enige **waarheid** is dat de journalisten logen.
'The only truth is that the journalists lied.'
0.15 Het is de **bedoeling** dat de maatregel afschrikkend werkt.
'It is the goal that the measure works as a deterrent.'
0.14 Het **voordeel** daarvan is dat je de zaken op een onafhankelijke manier kunt bekijken.
'The advantage of that is that you can view things in an independent manner.'
0.14 De **verwachting** is dat de Italiaanse troepen zeker tot 2005 zullen blijven.
'The expectation is that the Italian troops will stay at least until 2005.'
0.14 Maar het is de **vraag** of dat ooit gaat lukken.
'But it is the question whether that will ever work.'

Figure 3: Top 5 matches for the shell noun example *De hoofdzaak is dat je beter wordt* using the Flat encoding strategy. Shell nouns are highlighted in boldface.

Strategy	NP with PP PR	Shell nouns PR
Subtree	91.4 ± 11.3	97.3 ± 3.7
Thin	42.7 ± 27.1	98.3 ± 2.6
Flat	89.3 ± 12.7	96.1 ± 6.2

Table 1: Average percentile rank of exact hits returned by GrETEL for both queries, according to each encoding strategy.

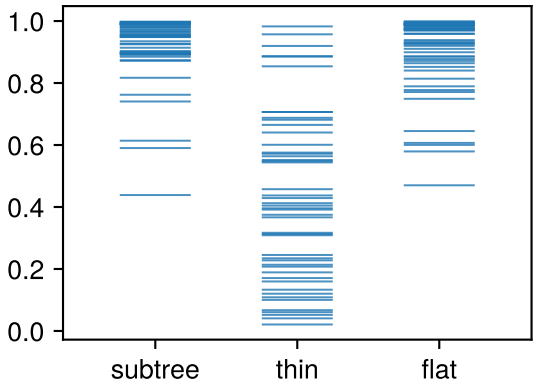


Figure 4: Percentile rank of exact hits for the NP with PP query, according to each encoding strategy.

The results for the first query, shown in Figure 4 are revealing: Here we observe a very wide spread of rankings for the exact matches when the Thin encoding strategy is used, which shows that there is little correlation between the rankings and the exact matches. The contrast is even more stark when we compare with the shell noun query, where no such deficit is evident. The contrast is probably due to the environments where each target pattern is found. Whereas shell nouns with complement

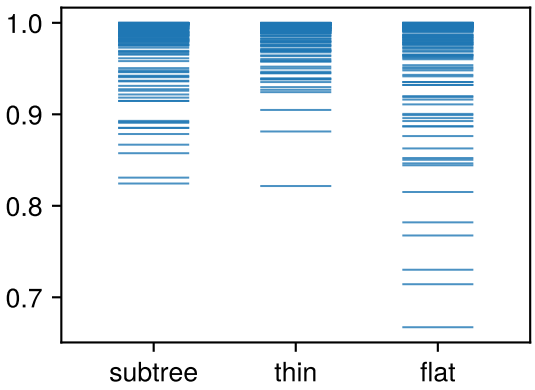


Figure 5: Percentile rank of exact hits for the shell noun query, according to each encoding strategy.

clauses are often or even usually found in matrix sentences, i.e. not embedded, NPs with PPs are likely to occur at a variety of depths in a parse tree. Since the query phrase is not embedded when parsed by Spacy and the Thin encoder only encodes full relation paths, the query and the stored sentence vectors are going to be dissimilar. This case demonstrates the utility of the Subtree and Flat encoding strategies.

Different kinds of data. One of the interesting possibilities that this class of approach offers, besides the computational simplicity and flexibility, is the ability to design representations that are both structured and support graded similarity. Whereas the default assumption that every symbol is different from all the other symbols would be reflected in the use of a new random vector for each symbol, it is also possible to apply various techniques to build in varying relations and degrees of similarity between vectors.

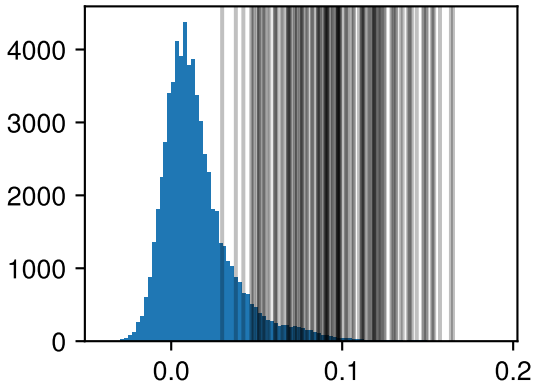


Figure 6: Overall distribution of similarities for the shell nouns query, using the Thin encoding strategy, with the similarity values for the exact hits superimposed as vertical lines.

For instance, in UD there are various subtypes of dependency relations. Thus you find relations like `acl:relcl` 'relative clause modifier', which is a subtype of the broader relation `acl` 'adnominal clause', and `aux:pass` 'passive auxiliary', which is a subtype of `aux` 'auxiliary'. In such cases, we can choose vectors for such subtypes that reflect their relatedness. For this, we use a function that returns a vector, in which some specified percentage of the elements are kept the same as a given input vector and the rest are random as usual. Thus the new vector will be the specified distance from the input vector and approximately distinct from other vectors generated in the same way.

The vectors for particular lexical items can be similarly adapted to evince a degree of similarity that is appropriate to a given application. Whereas a default approach would be to use random vectors for each new lexical item, saying merely that one item is different from another one and nothing more, one might derive lexical representations from various other sources. One could use word vectors derived from the (surface) distribution of words in a given corpus or one could use some kind of sensory data, such as imaging data, as in Eliasmith (2013), or even some combination of these.

The representations could also be designed to reflect properties of the word strings themselves. One way to encode strings is to assign random vectors to character *n*-grams as primitives and to compose them to arrive at the representation for a given string (Joshi et al., 2017). Another method uses "demarcator" vectors (Cohen et al., 2014) that correspond to each of the positions in the string, from left to right. These are each generated to be similar to their neighbors, such that a letter appearing close to the beginning of a string, whether at the second or third position, will have a similar vector representation. Both of these would differ more significantly from a letter bound to the demarcator vector corresponding to the end of a string.

It would also be simple to include information about each token's location in the sentence using this same demarcation vectors technique. Accordingly, tokens in a similar position relative to the beginning or end of a sentence would have greater similarity due to this feature's inclusion.

Limitations and open questions. While the fuzziness in the search results is automatic insofar as it follows directly from the properties of VSAs and it has certain advantageous qualities, e.g. that queries need not be precisely formulated and may consist of an example sentence as-is, there are also some potential disadvantages, specifically with regard to search applications.

In particular, it's not obvious how either certain parts of the query can be made into requirements or, by the same token, how certain results can be ruled out by way of constraints.

Instead of required query components, in Widdows and Cohen (2015) suggest "superposing additional cues" as one method of lending more weight to certain components of a query. So if it is important that the results include the word *fact*, then one would add $\rho^1 v_{\text{form}} \otimes v_{\text{fact}}$ two or three extra times into the query bundle. Then any sentences that include this feature would be ranked higher.

With respect to constraints, since non-matching elements only count as noise and thus have little effect on the similarity of a particular instance, it is also unclear how irrelevant instances can be made less similar so that they appear lower in the results ranking. It may be possible to include something like constraints by negating these irrelevant components (i.e., in BSC, flipping all the bits), $\rho^1 v_{\text{form}} \otimes v_{\text{worm}}^{-1}$ to make instances including the word form *worm* less similar, but this has not been tested yet.

6. Conclusion

In this paper we described an efficient means of making dependency treebanks easily and quickly searchable using techniques derived from vector symbolic architectures. The current application already provides a useful means of finding sentences that are syntactically similar to the provided query, whether the query consists of a complete example sentence or a phrase corresponding to some construction of interest. The implementation is quite simple conceptually, and it requires much less computational resources than comparable existing ap-

plications. Every query amounts to a simple vector operation that is performed exactly once for each sentence in the treebank.

Apart from being lightweight and simple to implement, this is an approach that has some intriguing properties, particularly in its fuzzy matching behavior and its ability to integrate a variety of kinds of data, thus we consider it to be worth further exploration.

As these initial tests have given some promising results, we plan to develop this approach further in future work. In particular, we plan to test the optional integration of word embeddings to represent an additional feature in each token bundle. The idea would be that syntactically similar sentences that also use distributionally similar vocabulary would be ranked higher than sentences that are similar in syntax alone. We also plan to determine whether it would be possible to transfer this approach to other kinds of VSAs, which, among other things, could make it easier to integrate embeddings.

7. Acknowledgments

This research is funded by the Deutsche Forschungsgemeinschaft (DFG) SFB 1475 – project number 441126958. We would like to thank Stefanie Dipper and the anonymous reviewers for their helpful comments and suggestions.

8. Bibliographical References

Liesbeth Augustinus, Vincent Vandeghinste, and Frank Van Eynde. 2012. Example-based treebank querying. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 3161–3167, Istanbul, Turkey. European Language Resources Association (ELRA).

Trevor Cohen, Dominic Widdows, Manuel Wahle, and Roger Schvaneveldt. 2014. *Orthogonality and Orthography: Introducing Measured Distance into Semantic Space*, pages 34–46. Springer Berlin Heidelberg.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, pages 1–54.

Chris Eliasmith. 2013. *How to Build a Brain: A Neural Architecture for Biological Cognition*. Oxford University Press.

Markus Gärtner, Gregor Thiele, Wolfgang Seeker, Anders Björkelund, and Jonas Kuhn. 2013. ICARUS – an extensible graphical search tool for dependency treebanks. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Sofia, Bulgaria. Association for Computational Linguistics.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength natural language processing in Python.

Aditya Joshi, Johan T. Halseth, and Pentti Kanerva. 2017. *Language Geometry Using Random Indexing*, page 265–274. Springer International Publishing.

Pentti Kanerva. 1996. *Binary spatter-coding of ordered K-tuples*, pages 869–873. Springer Berlin Heidelberg.

Pentti Kanerva. 2009. Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors. 1(2):139–159.

Jussi Karlgren. 2023. High-dimensional vector spaces can accommodate constructional features quite conveniently. In *Proceedings of the First International Workshop on Construction Grammars and NLP (CxGs+NLP, GURT/SyntaxFest 2023)*, pages 31–35, Washington, D.C. Association for Computational Linguistics.

Denis Kleyko, Dmitri A. Rachkovskij, Evgeny Osipov, and Abbas Rahimi. 2022. A survey on hyperdimensional computing aka. vector symbolic architectures, part I: Models and data transformations. *ACM Comput. Surv.*, 55(6):1–40.

Juhani Luotolahti, Jenna Kanerva, Sampo Pyysalo, and Filip Ginter. 2015. SETS: Scalable and efficient tree search in dependency graphs. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 51–55, Denver, Colorado. Association for Computational Linguistics.

Jan Odijk, Martijn van der Klis, and Sheean Spoel. 2017. Extensions to the GrETEL treebank query application. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories*, pages 46–55, Prague, Czech Republic.

Philip Resnik and Aaron Elkiss. 2005. The Linguist's Search Engine: An overview. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, pages 33–36, Ann Arbor, Michigan. Association for Computational Linguistics.

Kenny Schlegel, Peer Neubert, and Peter Protzel. 2022. A comparison of vector symbolic architectures. 55:4523–4555.

Hans-Jörg Schmid. 2000. *English Abstract Nouns as Conceptual Shells: From Corpus to Cognition*. De Gruyter.

Gertjan van Noord, Gosse Bouma, Frank Van Eynde, Daniël de Kok, Jelmer van der Linde, Ineke Schuurman, Erik Tjong Kim Sang, and Vincent Vandeghinste. 2012. *Large Scale Syntactic Annotation of Written Dutch: Lassy*, page 147–164. Springer Berlin Heidelberg.

D. Widdows and T. Cohen. 2015. Reasoning with vectors: A continuous model for fast robust inference. 23(2):141–173.

Modular Neural Machine Translation with a Semantic Pivot – Pilot Study Using AMR

Wenyang Gao[†*], **Yaxuan Li**[*], **Yunxin Bao**[†], **Shulin Huang**[*], **Yue Zhang**[*]
[†]Zhejiang University, [*]Westlake University
{gaowenyang, liyaxuan, huangshulin, zhangyue}@westlake.edu.cn
baoyunxin.cathy@outlook.com

Abstract

Neural machine translation (NMT) has become the predominant approach for automated translation, yet conventional models trained on extensive bilingual datasets exhibit critical limitations, including quadratic scaling of training data, sensitivity to out-of-distribution inputs, and a lack of interpretability. Inspired by the classical "translation pyramid" concept, which advocates for translation via a semantic pivot (interlingua), this work explores the integration of Abstract Meaning Representation (AMR) as a structured semantic intermediary to decouple translation into comprehension (source-to-AMR) and generation (AMR-to-target) phases. Concretely, we adopt English AMR as the pivot representation: all source sentences — regardless of their language — are mapped to English AMR graphs, which are then used to generate the target language. We conduct a pilot study using a strong AMR parser to create a multilingual silver-standard AMR corpus from the United Nations Parallel Corpus, covering six languages: Arabic (ar), Chinese (zh), English (en), French (fr), Russian (ru), and Spanish (es), training modular semantic understanding and generation components for each language. Experimental results demonstrate that our approach achieves an average improvement of 3% in robustness and over 15% in generalization compared to traditional Seq2Seq baselines. Analysis suggests that enhancing semantic parsing and generation accuracy could bridge the gap to conventional NMT systems. To our knowledge, this is the first work to integrate AMR as a semantic pivot in NMT, offering enhanced transparency, scalability, and robustness. This study underscores the potential of semantic-driven translation frameworks and provides a foundation for future research in interpretable, resource-efficient multilingual systems.

Keywords: machine translation, abstract meaning representation, semantic pivot

1. Introduction

Neural machine translation (NMT) (Bahdanau et al., 2014) has become the dominant method for machine translation. Traditional NMT methods use a neural string transducer architecture, typically the Transformer (Vaswani, 2017), which consists of a neural encoder and a neural decoder. The former is used to find a representation of the source input, while the latter generates the target output sequence. Trained on large datasets (e.g., tens of millions of bilingual sentence pairs), NMT models yield strong results for resource-rich languages. Recently, with advances in large language models (LLMs) (Touvron et al., 2023; Xu et al., 2023; Ganesh et al., 2025), decoder-only NMT systems have attracted increasing attention. These methods typically make use of a base LLM that contains knowledge across many languages and fine-tune the model with a relatively smaller amount of bilingual sentence pairs (e.g., tens of thousands). By leveraging the rich multilingual knowledge in base models, LLM-based MT has shown promising performance.

Despite their strong performance, existing NMT systems still face several fundamental challenges. First, they rely on extensive multilingual sentence pairs for training, which scales with square complexity (Koehn and Knowles, 2017). For traditional NMT systems, this can necessitate training $O(n^2)$ separate models. While modern multilingual NMT systems can consolidate this into a single universal model, the complexity and language balance of the training data remain a significant challenge(Johnson et al., 2016; Aharoni et al., 2019). Second, existing NMT systems have been shown to be sensitive to out-of-distribution (OOD) data (Yin et al., 2023; Li et al., 2024), and give low performances on low-resource language pairs. Spurious features have been shown to be frequent in sequence-to-sequence (Seq2Seq) transducers, which deviate from the human rationale for translation (Yin et al., 2022). Third, NMT models are not directly interpretable in their translation process, and thus, it can be difficult for monolingual speakers to identify potential errors (Belinkov and Glass, 2019).

One key cause of the above shortcomings is the Seq2Seq mapping nature of NMT models. They learn to synthesize the target translation conditioned on the source sequence token by token, rather than first comprehending the source and then yielding the target output according to the semantic representation (Bender et al., 2021). Surface-level mappings are shallow, require large bilingual training data, and can be prone to spurious features. As shown in Figure 1, in traditional machine translation research, a "translation pyramid" model has been discussed, which shows an ideal scenario that

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 74–85

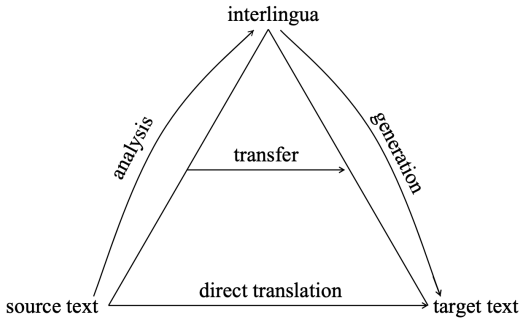


Figure 1: The machine translation pyramid categorizes approaches based on the extent of analysis and generation required. The interlingua approach involves comprehensive analysis and generation, while the direct translation approach minimizes both. The transfer approach falls between these two extremes.

makes use of a semantic pivot (or inter-lingua) for understanding-writing-style translation (Hutchins, 1995). A fundamental advantage of translating through a shared semantic representation lies in its inherent modularity and robustness.

The process involves: (i) Semantic Understanding: converting the source sentence into a language-independent meaning representation, and (ii) Surface Realization: generating the target sentence from this representation. This structure naturally avoids overfitting to superficial source-target correlations, leading to better generalization. Moreover, translation errors become interpretable—they stem either from inaccuracies in the semantic representation or from failures in the target language generation. Crucially, the modular design requires only O(n) components: each language necessitates one module for semantic encoding and one for surface generation, scaling efficiently to many languages.

The strong power of neural language models gives large potentials for building a translation system using a semantic pivot (Tenney et al., 2019). Unfortunately, little research has been done to this end in the literature (Matla et al., 2023). One reason may be that existing language models are trained over huge texts, yet it can be difficult to find semantic structures on the same scale (Raffel et al., 2019). In addition, there is no universally agreed semantic structure for cross-lingual representation. Despite the above challenges, some fundamental research in exploring the potential of semantic pivots can still provide valuable information. To this end, we take the Abstract Meaning Representation (AMR) (Banarescu et al., 2013) as the pivot and conduct a pilot study. Specifically, we adopt English AMR as the pivot throughout all translation directions. While AMR datasets for other languages do exist (e.g., Spanish, Chinese, and Czech AMR banks), their scale remains limited compared to English resources, and state-of-the-art AMR parsers are predominantly trained on English data. We therefore treat English AMR as a practical and well-supported choice for a first pivot study, while acknowledging that this introduces an inherent English-centric bias, a limitation we return to in Section 6.

Without aiming to immediately achieve competitive results to the state-of-the-art, we endeavor to answer the following research questions:

- Can we build a decent NMT system using existing semantic resources?
- Can a semantic pivot give more robust translation results compared to Seq2Seq transducer in the NMT context?
- With the huge potential space for enhancing semantic structures, can increase in the semantic parsing and semantic generation accuracies lead to strong translators?

In our experiments, we use a state-of-the-art AMR parser to annotate over 11 million data points in the United Nations Parallel Corpus (UN datasets), creating a multilingual silver AMR dataset, where we train our modules for each language. We combine theses semantic understanding modules and language generation modules into a complete multilingual neural machine translation (M-NMT) system, establishing a transparent pipeline architecture.

Our method shows an average improvement of 3% for each language pair in the robustness test, compared to the baseline model. In the generalization test, our model outperforms the baseline by over 15%. We also conduct a performance analysis, revealing that data quality is the main bottleneck. As data quality improves, our translation approach has the potential to achieve results comparable to, or even better than, traditional Seq2Seq models. To our knowledge, we are the first to incorporate AMR as a pivot in the context of NMT. Our semantic-understanding-based translation approach surpasses traditional models in robustness, generalization, and interpretability.

2. Related Work

Pivot-based NMT. Due to the lack of data in low-resource languages, conventional pivot-based NMT approaches (Johnson et al., 2017) enhance translations between low-resource languages by incorporating a high-resource language as a pivot. Some researchers (Leng et al., 2019) refine the process by adopting a learning-to-route (LTR) method to optimize the choices of the pivot languages and path from source to target. Another pivot-based

NMT paradigm involves generating more multilingual training data through data augmentation (Park et al., 2017; Currey and Heafield, 2019) or knowledge distillation (Chen et al., 2017; Ahmed and Buys, 2024), further fine-tuning the model to reduce the impact of data sparsity on model performance. Unlike previous methods, we alleviate the explicit semantic representation as a pivot and formulate the end-to-end translation into a human-like process involving a semantic understanding module and a language generation module. This human-like translation improves performance, robustness, and interoperability.

Transfer Learning in NMT. Transfer Learning (TL) is a subfield of Machine Learning that reuses knowledge from solving one task (the parent) to address a different but related task (the child) (Pan and Yang, 2010; Zhuang et al., 2021). Researchers fine-tune bilingual (Maimaiti et al., 2019; Maimaiti and Sun, 2020) or multilingual (Stickland et al., 2021) PLMs on low-resource language data to stimulate their translation capabilities. However, parallel data for low-resource languages are difficult to acquire and usually come from domains such as religious text (Siddhant et al., 2022). To transfer the abilities of multilingual PLMs, we fine-tune them on the AMR parsing task and AMR-to-text task respectively, acquiring the semantic understanding module and language generation module. In this process, we use monolingual data to eliminate reliance on low-resource parallel corpora.

3. Method

Our study proposes a modular semantic machine translation approach that introduces AMR as an intermediate semantic representation, which decomposes the translation task into two independent stages: semantic understanding and language generation. As shown in Figure 2, the semantic understanding module transforms text in the source language into an AMR graph, which is subsequently processed by the language generation module to generate text in the target language.

3.1. AMR as Pivot

AMR is a heavily abstracted graph-based representation of surface text, compressing semantics information into nodes and edges. Importantly, AMR was originally designed as an English-centric formalism: concept strings are grounded in English PropBank frames and vocabulary. In this work, we use English AMR exclusively as the semantic pivot, meaning that for all source languages, the semantic understanding module produces an English AMR graph, which the language generation module then realizes in the target language. This choice is motivated by the maturity of English AMR parsers and the relative abundance of English AMR-annotated data compared to other languages. Many researchers use AMR as a pivot in various NLP tasks, such as text paraphrasing (Fan and Gardent, 2020; Wang et al., 2020), text style transfer (Shi et al., 2023; Jangra et al., 2022), and translationese reduction (Wein and Schneider, 2023). Their studies prove that introducing the AMR-as-a-Pivot paradigm enhances performance and improves robustness for NLP tasks.

Seq2Seq models, while effective for sequential data, often fall short when dealing with the structural complexity of natural language. To address this, we leverage the depth-first search (DFS) algorithm, which aligns well with the linearized syntactic structures of natural language (Bevilacqua et al., 2021). For instance, the AMR graph in Figure 2 is linearized as: `( <pointer:0> remain-01 :ARG1 ( <pointer:1> incident :quant 1 ) :ARG3 ( <pointer:2> verify-01 :ARG1 <pointer:1> ) )`, where the `<pointer:i>` values are node coreference indices. To process AMR symbols effectively, we expand the vocabulary to include all relevant relations and frames. Moreover, we introduce a novel language token, `<amr_XX>`, to the existing language code. This allows multilingual models to seamlessly recognize and generate AMR as a pivot language, facilitating efficient cross-lingual translation.

3.2. Semantic Understanding Module

The purpose of the semantic understanding module is to extract and represent semantic information from the source language text as a structured AMR graph. This module can: (1) map surface-level words to specific nodes based on contextual clues, effectively handling polysemy and synonymy; and (2) represent semantic relationships between concepts as edges connecting nodes within the AMR graph. These capabilities are grounded in the ability to process structured sequences, as demonstrated by Transformer models, enabling the module to effectively capture the underlying semantic structure of the text.

In our proposed method, the semantic understanding module consists of an encoder and a decoder. The encoder is primarily tasked with reading the input sentence in natural language and transforming it into a high-dimensional dense vector representation that encapsulates the sentence's semantic content. Typically implemented as a Transformer, the encoder processes input word embeddings through multiple neural network layers, employing self-attention mechanisms to capture dependencies among words. Moreover, the encoder benefits from pre-training on large multilingual cor-

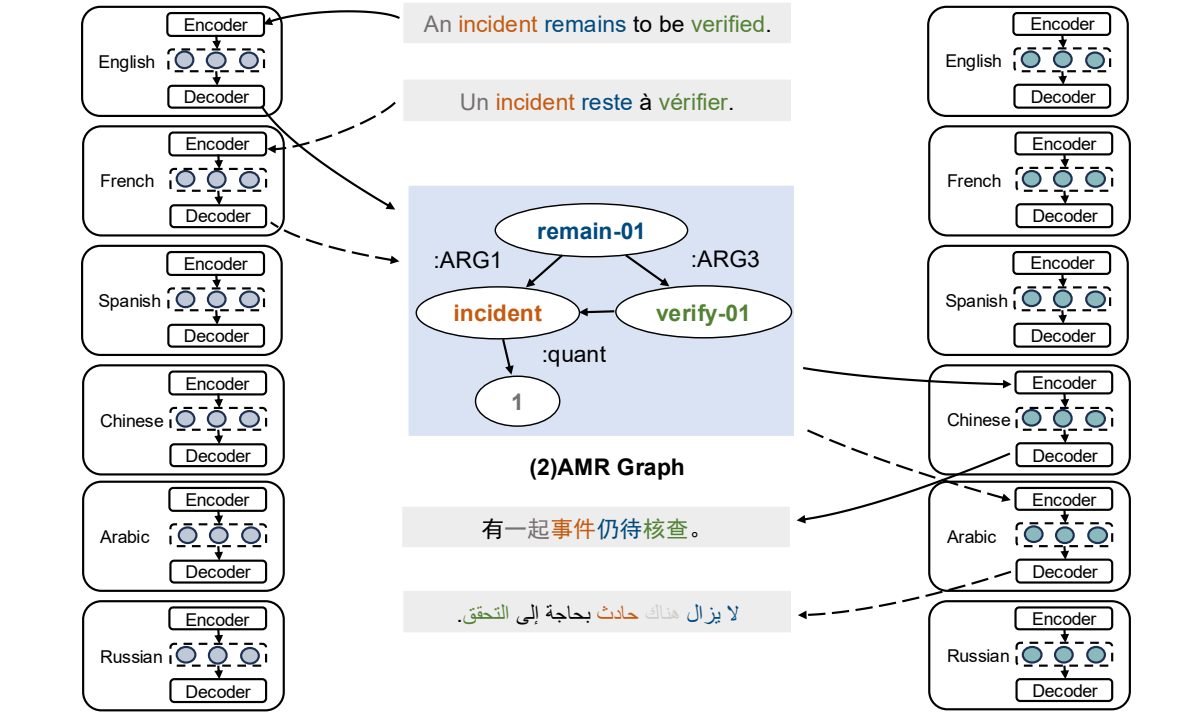


Figure 2: **Overview of the Proposed AMR-Pivot M-NMT Pipeline.** The pipeline illustrates the translation process for both English-to-Chinese (en-zh) and French-to-Arabic (fr-ar) pairs. Words in different languages that convey the same semantic meaning are highlighted with the same color, emphasizing alignment achieved through AMR pivot.

pora, which enables it to acquire cross-linguistic semantic features and enhances its generalizability across diverse source languages.

The decoder, in turn, converts the semantic vector representation generated by the encoder into a linearized AMR graph sequence. This sequence represents a directed graph where nodes correspond to concepts, and edges denote relationships between them. During training, the decoder's parameters are optimized by aligning its output with the ground truth AMR linear sequence, often through minimizing cross-entropy loss or similar loss functions suited to sequence generation tasks.

3.3. Language Generation Module

The language generation module aims to produce appropriate text in the target language based on the given AMR graph, ensuring that the generated translation is consistent with the semantic meaning of the source text.

The encoder's primary function is to transform the input AMR graph, which represents the underlying semantic structure through nodes (concepts) and edges (relationships between concepts), into a dense intermediate representation. This transformation is critical for capturing both the structure and meaning embedded in the AMR. In the language generation module, the encoder processes a linearized form of the AMR graph through self-attention mechanisms to capture the semantic relationships within the graph, thus producing a continuous vector representation that encapsulates the entire semantic meaning of the graph, which is subsequently processed by the decoder to generate text in the target language.

The decoder is tasked with converting the encoded semantic representation into a fluent sequence of text in the target language. It must ensure that the generated text accurately reflects the meaning represented by the AMR while adhering to the linguistic and syntactic rules of the target language. Typically integrated into a Transformer architecture, the decoder generates the target text autoregressively, token by token, by utilizing both the encoded semantic information and previously generated tokens. In multilingual settings, the decoder must accommodate cross-linguistic syntactic differences and word order variations to ensure natural and fluent output in the target language.

3.4. Modular Interaction

Here, we denote S as the source language, T as the target language, and AMR as the semantic representation.

- **Semantic Understanding Module**. The purpose of this module is to extract the semantic meaning from the source language text and convert it into a serialized AMR graph, which serves as the intermediate representation during the translation process $(S \rightarrow AMR)$.
- **Language Generation Module.** This module aims to produce appropriate text in the target language based on the given AMR graph, ensuring that the generated translation is consistent with the semantic meaning of the source text $(AMR \rightarrow T)$.

We denote $\mathcal{M}^{S}_{AMR}$ as the semantic understanding module for source language S, and $\mathcal{M}^{AMR}_{T}$ as the language generation module for the target language T. For each language, we fine-tune both modules on the corresponding data, resulting in a list of semantic understanding modules $\{\mathcal{M}^{S}_{AMR} \mid S \in \{ar, zh, en, fr, ru, es\}\}$ and a list of language generation modules $\{\mathcal{M}^{AMR}_{T} \mid T \in \{ar, zh, en, fr, ru, es\}\}$. Both modules are optimized using cross-entropy loss:

$$\mathcal{L}_{I,O} = -logP(\widehat{O}|I, O) \quad (1)$$

where (I, O) is the input and reference output for the module, and $(I, O) \in \{(S, AMR), (AMR, T)\}$. $\widehat{O}$ denotes the module-generated output.

As shown in Figure 2, this architecture allows for a flexible, modular approach to multilingual translation. A language-specific semantic understanding module and a language generation module can be combined to perform translation between any of the supported languages $(\mathcal{M}^{S}_{AMR} \oplus \mathcal{M}^{AMR}_{T})$. This process also yields an explicit semantic representation, which improves interpretability $(S \rightarrow AMR \rightarrow T)$.

4. Experiments

4.1. Datasets and Data Construction

We construct an extensive multilingual parallel dataset (UN-AMR) consisting of six official languages of the United Nations: Arabic, Chinese, English, French, Russian, and Spanish. We use the United Nations Parallel Corpus (Ziemski et al., 2016) and follow its original data split. We generate silver AMR annotations by parsing the corresponding English text using AMRBART (Bai et al., 2022). These AMR graphs serve as explicit semantic representations for texts in each language, paired with sentences in respective languages to form a parallel corpus with six subsets:

$$\mathcal{D} = \{(ar, AMR), (zh, AMR), (en, AMR), (fr, AMR), (ru, AMR), (es, AMR)\}$$

To assess the performance of our proposed modular semantic machine translation framework across varying data scales, we conducted comparative experiments using both the complete dataset and a subset from the UN-AMR corpus training partition. We denote the complete dataset as UN-AMR and the subset as UN-AMR-subset. The UN-AMR-subset consists of the first 640K instances of the complete dataset. The UN-AMR dataset is used to train models from scratch, while the UN-AMR-subset is employed for fine-tuning pre-trained models. The dataset statistics are detailed in Table 1. For each experimental condition, we executed parallel training procedures for both our model and baseline architectures, followed by comprehensive evaluations of their generalization capability and robustness across the corresponding dataset.

	#Sentences	
Split	**UN-AMR**	**UN-AMR-subset**
Train	11,365,709 (11.4M)	640,000 (640K)
Test	4,000	
Validation	4,000	

Table 1: Statistics of UN-AMR dataset.

4.2. Metrics

We evaluate translation quality using BLEU (Papineni et al., 2002) and COMET (Rei et al., 2020) on the standard UN-AMR test set and paraphrased versions to measure robustness. Additionally, we employ Smatch (Cai and Knight, 2013a) to evaluate the semantic accuracy of our AMR graph generation, comparing generated graphs against silver annotations.

BLEU. This metric is used to evaluate the quality of machine-translated text by comparing it to human-translated references. It calculates the n-gram precision between the machine-translated text and the reference translations. We evaluate the robustness of the translation system by measuring the rate of decline in BLEU scores.

COMET. A neural network-based metric for machine translation evaluation predicts human judgments by leveraging pre-trained models fine-tuned on human evaluation data. The model-based metric, COMET, offers a semantic-level assessment of machine translation quality.

Smatch. An evaluation metric specifically designed for semantic feature structures. It calculates a similarity score between two AMR graphs based on their structural and lexical similarities (Cai and Knight, 2013b). To evaluate our understanding modules' ability to capture semantic features, we

	AMRpivot on testset						AMRpivot on paraphrased testset					
$S \backslash T$	**ar**	**en**	**es**	**fr**	**ru**	**zh**	**ar**	**en**	**es**	**fr**	**ru**	**zh**
ar	-	37.0	35.8	27.3	25.0	38.6	-	29.2 $_{\downarrow 21\%}$	30.9 $_{\downarrow 14\%}$	23.6 $_{\downarrow 14\%}$	21.0 $_{\downarrow 16\%}$	33.9 $_{\downarrow 12\%}$
en	22.4	-	42.8	32.0	28.7	43.1	15.4 $_{\downarrow 31\%}$	-	32.1 $_{\downarrow 25\%}$	25.2 $_{\downarrow 21\%}$	22.2 $_{\downarrow 23\%}$	35.9 $_{\downarrow 17\%}$
es	19.9	41.7	-	30.6	26.3	39.3	16.5 $_{\downarrow 17\%}$	33.2 $_{\downarrow 20\%}$	-	27.1 $_{\downarrow 11\%}$	23.2 $_{\downarrow 12\%}$	35.8 $_{\downarrow 9\%}$
fr	17.4	35.7	35.9	-	24.3	36.2	14.6 $_{\downarrow 16\%}$	29.3 $_{\downarrow 18\%}$	31.6 $_{\downarrow 11\%}$	-	21.2 $_{\downarrow 12\%}$	33.3 $_{\downarrow 8\%}$
ru	17.9	35.3	34.1	27.2	-	36.3	14.7 $_{\downarrow 18\%}$	28.2 $_{\downarrow 20\%}$	29.8 $_{\downarrow 13\%}$	23.9 $_{\downarrow 12\%}$	-	32.7 $_{\downarrow 10\%}$
zh	17.1	34.4	33.1	25.3	23.4	-	14.4 $_{\downarrow 16\%}$	28.6 $_{\downarrow 17\%}$	29.5 $_{\downarrow 11\%}$	23.0 $_{\downarrow 9\%}$	20.7 $_{\downarrow 12\%}$	-
Avg↓	5.89 (16%)											

Table 2: Robustness results of our proposed model.

	Transformer on testset						Transformer on paraphrased testset					
$S \backslash T$	**ar**	**en**	**es**	**fr**	**ru**	**zh**	**ar**	**en**	**es**	**fr**	**ru**	**zh**
ar	-	42.95	40.48	28.26	28.35	39.0	-	31.7 $_{\downarrow 26\%}$	32.9 $_{\downarrow 19\%}$	25.4 $_{\downarrow 10\%}$	22.8 $_{\downarrow 20\%}$	35.1 $_{\downarrow 10\%}$
en	28.52	-	55.10	41.01	36.60	47.7	18.0 $_{\downarrow 37\%}$	-	37.6 $_{\downarrow 32\%}$	30.1 $_{\downarrow 27\%}$	25.3 $_{\downarrow 31\%}$	38.8 $_{\downarrow 19\%}$
es	21.96	45.41	-	39.02	31.61	42.3	18.1 $_{\downarrow 18\%}$	34.9 $_{\downarrow 23\%}$	-	31.6 $_{\downarrow 19\%}$	26.5 $_{\downarrow 16\%}$	38.8 $_{\downarrow 8\%}$
fr	19.31	43.49	45.52	-	29.87	24.6	16.1 $_{\downarrow 17\%}$	34.1 $_{\downarrow 22\%}$	36.7 $_{\downarrow 19\%}$	-	25.0 $_{\downarrow 16\%}$	22.3 $_{\downarrow 9\%}$
ru	16.87	41.96	40.06	30.93	-	40.3	14.5 $_{\downarrow 14\%}$	32.6 $_{\downarrow 22\%}$	32.5 $_{\downarrow 19\%}$	25.3 $_{\downarrow 18\%}$	-	36.4 $_{\downarrow 10\%}$
zh	18.49	41.02	38.15	30.56	27.49	-	15.2 $_{\downarrow 18\%}$	32.3 $_{\downarrow 21\%}$	31.3 $_{\downarrow 18\%}$	26.0 $_{\downarrow 15\%}$	23.0 $_{\downarrow 16\%}$	-
Avg↓	8.15 (19%)											

Table 3: Robustness results of baseline model.

calculate Smatch scores by comparing their generated AMR graphs against silver AMR annotations.

Human Evaluation. To supplement automatic metrics, we conduct human evaluation for en→zh and zh→ar. Bilingual annotators rate each translation on adequacy and fluency. Inter-annotator agreement is reported via Cohen's κ.

4.3. Baseline Models

We adopt mBART-cc25-large (hereafter referred to as mBART-25) (Liu et al., 2020) as the foundational architecture for our experiments. We fine-tune mBART-25 on the UN-AMR-subset. Both the semantic understanding module and the language generation module are based on this architecture in our proposed framework. Our AMR-pivot translation system is denoted as `AMRpivot`.

For a rigorous comparison, we also employ mBART-25 as the foundational model for the baseline system. We develop an M-NMT system comprising 30 models, each specifically tailored to a distinct language pair. These models are trained on datasets identical to those used in our method for the respective language pairs with source text as input and target text as output. Similar to our proposed method, the baseline model is denoted as `Transformer`.

4.4. Robustness Test

We evaluate the robustness of each translation framework by measuring how sensitive its output quality is to surface-level variation in the input. Specifically, we use GPT-4o[1] to paraphrase each source sentence in the test set while preserving its semantics[2], then translate both the original and paraphrased sentences using the fine-tuned models. BLEU scores are computed for both sets of translations against the original reference translations, and robustness is quantified as the **percentage decline in BLEU score** between the original and paraphrased inputs.

Why percentage decline, not absolute BLEU score. A critical consideration when interpreting these results is the asymmetry between `AMRpivot` and `Transformer` in terms of pre-trained knowledge. Both systems are built on mBART-25, which has been pre-trained on large multilingual corpora and already encodes substantial translation knowledge for the six UN languages. When fine-tuned directly on source–target sentence pairs, the baseline `Transformer` can leverage this pre-existing translation competence, leading to higher absolute BLEU scores on the standard test set. By contrast, our system `AMRpivot` fine-tunes mBART-25 on the AMR parsing and AMR-to-text tasks, neither of which directly exercises the model's translation knowledge. As a result, the two systems begin from meaningfully different effective starting points: `Transformer` benefits from a head start in bilingual signal, while `AMRpivot` must construct its translation capability almost entirely through the AMR intermediate representation.

In this context, comparing absolute BLEU scores conflates two different things: the quality of the un-

[1]The version employed is gpt-4o-2024-08-06.

[2]Prompt: "Please rewrite this sentence while keeping the semantics as unchanged as possible."

	BLEU		COMET	
	Random-test	CG-test	Random-test	CG-test
mBART-25	61.4	58.0 $_{\downarrow 6\%}$	72.7	63.0 $_{\downarrow 13\%}$
Transformer	51.2	43.2 $_{\downarrow 16\%}$	62.0	49.9 $_{\downarrow 20\%}$
AMRpivot	38.6	37.9 $_{\downarrow 2\%}$	47.4	47.5

Table 4: Generalizability on CoGnitnion.

derlying translation framework and the degree to which the model can exploit pre-trained bilingual knowledge. What we are specifically interested in here is the former — how much a framework's output degrades when the surface form of the input changes while meaning is preserved. A system that genuinely translates via semantic understanding should, in principle, be invariant to such surface variation, whereas a system that relies on surface co-occurrence patterns should degrade more. The percentage decline in BLEU score isolates exactly this property, making it the appropriate primary metric for robustness comparison. We nonetheless report absolute scores in full in Table 2 and 3 for completeness, and note that even on this metric, AMRpivot outperforms Transformer on certain language pairs (e.g., ru–ar and fr–zh) despite the inherent disadvantage described above.

Overall, our method achieves an average BLEU decline of 16% across language pairs on paraphrased inputs, compared to 19% for the Transformer baseline — an average improvement of 3 percentage points per language pair. As shown in Tables 2 and 3, the advantage is particularly pronounced for language pairs involving Arabic and Russian, which are relatively lower-resourced within mBART-25's pre-training data, suggesting that the semantic pivot provides the most benefit precisely where surface-level pre-trained knowledge is weakest.

4.5. Generalizability Test

Further, to better validate our conclusion that the translation framework we propose relies more on semantic representation than surface sentence structures compared to the traditional translation framework, we design a two-stage generalization test. In the first stage, based on the experimental results from Section 4.4, we selected the English-to-Chinese language direction (which achieved the highest BLEU score in both translation models) for verification. We train two models, namely Transformer and AMRpivot, from scratch using the entire UN-AMR dataset, consuming 80*A100 GPU hours of computational resources. It is important to note that, according to the scaling law, when the training data scale reaches the computational optimal boundary, the model performance improvement enters a plateau phase. The experimental configuration is nearly at this critical point, which ensures that the results effectively reflect the inherent characteristics of the model framework.

We adopt the CoGnition (Li et al., 2021) dataset for evaluation of generalizability. The dataset contains 216,246 training sentence pairs, and the test set is composed of two subsets: (1) the conventional test set (10,000 sentence pairs), which contains typical combinations of training vocabulary; (2) the combination generalization test set (CG-testset, 10,800 sentence pairs), which specifically constructs 2,160 novel compounds, with up to 5 atoms and 7 words. We fine-tuned the two models obtained in the previous step on the training dataset and tested them on both two subsets to distinguish the model's memory of surface structures from its generalization ability to deep semantics.

The comparative experimental results on both test subsets are shown in Table 4. As shown, our model performs almost identically on both the CG-test and the conventional test set, with even a slight increase in BLEU score, while other benchmark models show significant declines. This demonstrates that our translation framework enables the model to learn deeply from semantics, so the combination generalization of the data does not significantly affect model performance. It is worth mentioning that since the AMR used for training was derived from AMR parser-generated silver data, which has lower precision, the large dataset (11M instances) leads to significant error accumulation during the semantic understanding process. Therefore, directly comparing the absolute results would not be fair.

Beyond absolute performance, the contrast between AMRpivot and direct transduction models on the CG-test set reflects differences in inductive bias. The combination generalization split introduces novel lexical compounds unseen during training. For Seq2Seq models, translation quality partly relies on memorized surface co-occurrence patterns, and novel combinations therefore cause performance degradation.

In contrast, the AMR-based framework separates lexical realization from predicate-argument structure. The semantic understanding module maps surface expressions into structured graphs capturing events and relations, allowing many unseen lexical combinations to correspond to previously observed semantic substructures composed in new ways. Let $\mathcal{X}$ denote the surface string space and $\mathcal{Z}$ the semantic graph space. While $\mathcal{X}$ grows combinatorially with vocabulary size, $\mathcal{Z}$ consists of structured recombinations of a finite predicate and role inventory. Modeling translation via $\mathcal{Z}$ thus compresses the hypothesis space and promotes systematic recombination rather than memorization of surface patterns.

The near-identical performance of AMRpivot on

ID	Source	AMR	Translation	Reference
1	送往Shami医院的**孩子名单**	(<pointer:0> thing :ARG2-of (<pointer:1> list-01 :ARG1 (<pointer:2> person :ARG0-of (<pointer:3> have-rel-role-91 :ARG2 (**<pointer:4> client**)) :ARG1-of (<pointer:5> send-01 :ARG2 (<pointer:6> hospital :wiki - :name (<pointer:7> name :op1 <lit> Shami </lit> :op2 <lit> Hospital </lit>))))))	**Lists of clients** sent to Shami Sham Hospital.	**Lists of children** taken to Shami Hospital.
2	它是玻璃**做的**。	(<pointer:0> **make-01** :ARG1 (<pointer:1> it) :ARG2 (<pointer:2> glass))	It was **making** glasses of it.	It was **made of glass**.
3	在取消了对欧元汇率**下限**的同时，还将活期存款账户余额利率降为负数，为-0.75%		In addition to the removal of the **ceiling** on the exchange rate for the Euro, the interest rate on the balance of deposit accounts would be reduced to a negative level of -0.75 percent.	The removal of the **floor** with respect to the euro was accompanied by a further move into negative territory of the interest rate on sight deposit account balances to -0.75 percent.

Figure 3: Mistranslation instances for the outputs of two translation systems.

both test sets supports this interpretation, suggesting that explicit semantic abstraction provides an effective inductive bias for compositional generalization.

4.6. Human Evaluation

To complement the automatic metrics, we conduct a small-scale human evaluation assessing adequacy (meaning preservation) and fluency (grammaticality and naturalness), each rated on a 1–5 scale.

We focus on en→zh and zh→ar. For each language pair, we randomly sample 25 sentences from the UN-AMR test set, each evaluated in both its original and paraphrased form (using the same GPT-4o procedure as Section 4.4), yielding 50 translation instances per system. The two translations per sentence are presented to two bilingual annotators in randomized order without system labels, producing 200 judgments per dimension per language pair (25 × 2 conditions × 2 systems × 2 annotators). Disagreements are resolved by averaging, and inter-annotator agreement is reported via Cohen's κ.

As shown in Table 5, `AMRpivot` achieves higher adequacy on both language pairs under both conditions, with the advantage widening on paraphrased inputs, from $+0.14$ to $+0.31$ on en→zh and from $+0.13$ to $+0.27$ on zh→ar, which directly confirming the robustness findings of Section 4.4. `Transformer` scores higher on fluency throughout, reflecting its direct exposure to source–target pairs during fine-tuning. The fluency gap is larger on zh→ar (-0.54) than en→zh (-0.37), consistent with greater error accumulation through the AMR parsing stage on a more distant language pair. This adequacy–fluency tradeoff points to improving the language generation module — through higher-quality AMR annotations or stronger generative models — as the primary direction for future work.

Condition	System	en→zh		zh→ar	
		Adequacy	Fluency	Adequacy	Fluency
Original	`Transformer`	3.74	**3.98**	3.31	**3.52**
	`AMRpivot`	**3.88**	3.61	**3.44**	2.98
Paraphrased	`Transformer`	3.49	**3.74**	3.05	**3.16**
	`AMRpivot`	**3.80**	3.21	**3.32**	2.60
Cohen's κ		0.66	0.72	0.63	0.69

Table 5: Human evaluation results (mean scores on a 1–5 scale) for en→zh and zh→ar.

5. Interpretability Analysis

Figure 3 displays three instances of mistranslation. The first two examples are produced by our proposed model, while the third is generated by the Transformer baseline model. As shown, the first two cases indicate the effectiveness of our method in localizing error sources, distinguishing whether issues stem from the semantic understanding module or the language generation module, with AMR serving as a pivot. In contrast, the final example, generated by the traditional NMT system, fails to explicitly identify whether the errors originate from the encoder or decoder components.

For the first instance, according to AMR annotation conventions, `ARG1` typically represents the direct object of an action, while `ARG2` is often used for instrumental or comitative relations. In this case, Chinese characters meaning "Lists of children" are mistranslated as "Lists of clients". With the help of AMR pivot, the error can be easily identified. This

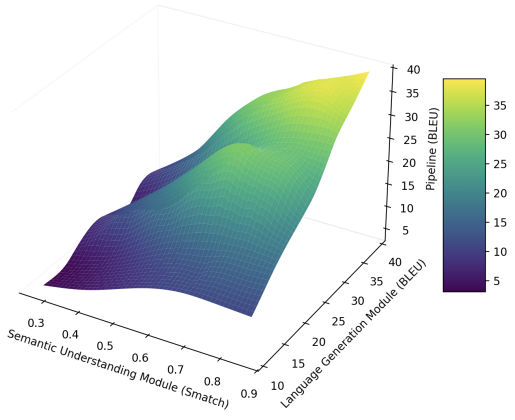


Figure 4: Overall translation performance of AMR-pivot with sacling abilities of semantic understanding modules and language generation modules..

mistake suggests that the model may have misinterpreted the Chinese characters meaning “children” during the semantic understanding process. By correcting the AMR parse structure, the model successfully generates the correct translation. In the second instance, the AMR graph is correctly extracted but it erroneously realizes the concept “make-01” as “making” during the language generation process and finally generates an incorrect target sequence.

In the third instance, Chinese characters meaning “floor” was erroneously translated as “ceiling” rather than “floor”. However, it is unclear where the error originates. Specifically, it is uncertain whether the mistake arises from a misinterpretation of the source language or an error in generating the target language. This demonstrates that the transformer model is unable to effectively identify such errors. This comparative analysis underscores an important distinction: While Transformer models rely on implicit vector representations that obscure error localization, our approach allows for explicit identification of error sources within specific processing modules. The opaque nature of intermediate representations in Transformer architectures complicates the diagnosis of translation errors, making it difficult to ascertain whether the issue arises from semantic misinterpretation or generation failures, thereby hindering targeted error correction. Moreover, this interpretability also lays the foundation for us to selectively improve the overall performance.

6. Challenges and Opportunities

We also train semantic understanding modules and language generation modules of varying capacities using different sizes of training datasets, then combine them pairwise into a pipeline to evaluate translation performance.

As shown in Figure 4, the performance of the AMR-based model is currently limited by the effectiveness of the semantic understanding module and the language generation module. With improvements in these two key components, we anticipate a gradual performance enhancement of AMRpivot. Specifically, as these modules advance, the translation performance will continue to improve.

Additionally, since we use English AMR as the pivot, the system carries an inherent English-centric bias. For non-English source languages, the semantic understanding module must bridge not only a linguistic gap but also a conceptual one, as AMR frames and concept strings are grounded in English lexical and syntactic conventions. This may disproportionately disadvantage morphologically rich or syntactically distant languages such as Arabic and Russian. While language-specific AMR banks exist for a small number of languages, their scale and parser maturity currently preclude their use at the data volumes required here. Future work should explore whether a more language-neutral semantic representation or a cross-lingual AMR formalism that decouples concepts from English vocabulary — could alleviate this bias and further boost translation quality across typologically diverse language pairs. Such a representation should focus on compressing redundant information, making it adaptable to various languages. This shift would not only enhance the interpretability of the model but also significantly boost its translation performance across multiple languages. Moreover, if language models are capable of computing rich, high-level structures rather than merely fragmented token-level representations, and if these structures can be aligned across languages, then the translation paradigm that operates without bilingual training data is likely to achieve significant progress.

7. Conclusion

We conducted a pilot study on using a semantic pivot to achieve neural machine translation without sequence-to-sequence transduction learning, finding that it has the potential to reduce the influence of spurious features, leading to more robust, interpretable, and less bilingual-data-hungry neural models. Additionally, although the existing method does not yet compete with state-of-the-art NMT systems, there is a salient trend: improving semantic parsing and generation accuracies leads to stronger translation quality. Such findings can shed light on further research on the new translation paradigm. Future research directions include finding a strong universal semantic representation and discovering new foundational models that can provide reliable text-level semantic information, rather than relying on token-level vectors.

8. Acknowledgements

This work has been financially supported by the National Key R&D Program of China (Grant No. 2022YFE0204900) and its Macao counterpart funded by FDCT, Macao SAR (Grant No. FDCT/0070/2022/AMJ).

9. Bibliographical References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. *ArXiv*, abs/1903.00089.

Khalid Ahmed and Jan Buys. 2024. Neural machine translation between low-resource languages with synthetic pivoting. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pages 12144–12158. ELRA and ICCL.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Xuefeng Bai, Yulong Chen, and Yue Zhang. 2022. Graph pre-training for AMR parsing and generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 6001–6015. Association for Computational Linguistics.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In *LAW@ACL*.

Yonatan Belinkov and James Glass. 2019. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 610–623, New York, NY, USA. Association for Computing Machinery.

Michele Bevilacqua, Rexhina Blloshmi, and Roberto Navigli. 2021. One SPRING to rule them both: Symmetric AMR semantic parsing and generation without a complex pipeline. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12564–12573.

Shu Cai and Kevin Knight. 2013a. Smatch: an evaluation metric for semantic feature structures. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 2: Short Papers*, pages 748–752. The Association for Computer Linguistics.

Shu Cai and Kevin Knight. 2013b. Smatch: an evaluation metric for semantic feature structures. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 748–752.

Yun Chen, Yang Liu, Yong Cheng, and Victor O. K. Li. 2017. A teacher-student framework for zero-resource neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1925–1935. Association for Computational Linguistics.

Anna Currey and Kenneth Heafield. 2019. Zero-resource neural machine translation with monolingual pivot data. In *Proceedings of the 3rd Workshop on Neural Generation and Translation@EMNLP-IJCNLP 2019, Hong Kong, November 4, 2019*, pages 99–107. Association for Computational Linguistics.

Angela Fan and Claire Gardent. 2020. Multilingual amr-to-text generation. *arXiv preprint arXiv:2011.05443*.

Devalla Bhaskar Ganesh, Paruchuri Eesha Chowdary, Dokuparthi Nilesh, Jonnala HarshaVardhan Reddy, Chittela Venkata Sai Tarun Reddy, and Suryakanth V. Gangashetty. 2025. Advances in machine translation: A comprehensive survey of large language models. *2025 3rd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT)*, pages 1671–1675.

W. John Hutchins. 1995. Machine translation: A brief history.

Anubhav Jangra, Preksha Nema, and Aravindan Raghuveer. 2022. T-star: Truthful style transfer using amr graph as intermediate representation. *arXiv preprint arXiv:2212.01667*.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Z. Chen, Nikhil Thorat, Fernanda B. Viégas, Martin Wattenberg, Gregory S. Corrado, Macduff Hughes, and Jeffrey Dean.

2016. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda B. Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Trans. Assoc. Comput. Linguistics*, 5:339–351.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Yichong Leng, Xu Tan, Tao Qin, Xiang-Yang Li, and Tie-Yan Liu. 2019. Unsupervised pivot translation for distant languages. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 175–183. Association for Computational Linguistics.

Yafu Li, Yongjing Yin, Yulong Chen, and Yue Zhang. 2021. On compositional generalization of neural machine translation. *arXiv preprint arXiv:2105.14802*.

Yafu Li, Huajian Zhang, Jianhao Yan, Yongjing Yin, and Yue Zhang. 2024. What have we achieved on non-autoregressive translation? *ArXiv*, abs/2405.12788.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Mieradilijiang Maimaiti, Yang Liu, Huanbo Luan, and Maosong Sun. 2019. Multi-round transfer learning for low-resource NMT using multiple high-resource languages. *ACM Trans. Asian Low Resour. Lang. Inf. Process.*, 18(4):38:1–38:26.

Yang Liu Huanbo Luan Mieradilijiang Maimaiti and Maosong Sun. 2020. Enriching the transfer learning with pre-trained lexicon embedding for low-resource neural machine translation. *Tsinghua Science & Technology*, page 1.

Syed Matla, Ul Qumar, Muzaffar Azim, and S. M. K. Quadri. 2023. Neural machine translation: A survey of methods used for low resource languages. *2023 10th International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 1640–1647.

Sinno Jialin Pan and Qiang Yang. 2010. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.*, 22(10):1345–1359.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318. ACL.

Jaehong Park, Jongyoon Song, and Sungroh Yoon. 2017. Building a neural machine translation system using only synthetic parallel data. *CoRR*, abs/1704.00253.

Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. Comet: A neural framework for mt evaluation. *arXiv preprint arXiv:2009.09025*.

Kaize Shi, Xueyao Sun, Li He, Dingxian Wang, Qing Li, and Guandong Xu. 2023. Amr-tst: Abstract meaning representation-based text style transfer. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4231–4243.

Aditya Siddhant, Ankur Bapna, Orhan Firat, Yuan Cao, Mia Xu Chen, Isaac Caswell, and Xavier Garcia. 2022. Towards the next 1000 languages in multilingual machine translation: Exploring the synergy between supervised and self-supervised learning. *CoRR*, abs/2201.03110.

Asa Cooper Stickland, Xian Li, and Marjan Ghazvininejad. 2021. Recipes for adapting pre-trained monolingual and multilingual models to machine translation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 3440–3453. Association for Computational Linguistics.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. Bert rediscovers the classical nlp pipeline. In *Annual Meeting of the Association for Computational Linguistics*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée

Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *ArXiv*, abs/2302.13971.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Tianming Wang, Xiaojun Wan, and Hanqi Jin. 2020. Amr-to-text generation with graph transformer. *Transactions of the Association for Computational Linguistics*, 8:19–33.

Shira Wein and Nathan Schneider. 2023. Lost in translationese? reducing translation effect using abstract meaning representation. *arXiv preprint arXiv:2304.11501*.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2023. A paradigm shift in machine translation: Boosting translation performance of large language models. *ArXiv*, abs/2309.11674.

Yongjing Yin, Yafu Li, Fandong Meng, Jie Zhou, and Yue Zhang. 2022. Categorizing semantic representations for neural machine translation. In *International Conference on Computational Linguistics*.

Yongjing Yin, Jiali Zeng, Yafu Li, Fandong Meng, Jie Zhou, and Yue Zhang. 2023. Consistency regularization training for compositional generalization. In *Annual Meeting of the Association for Computational Linguistics*.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. 2021. A comprehensive survey on transfer learning. *Proc. IEEE*, 109(1):43–76.

Michal Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The united nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016*. European Language Resources Association (ELRA).

Gutenberg+: A More Temporally Faithful Corpus for Diachronic NLP

Leon Hammerla, Alexander Mehler
Goethe University Frankfurt, Text Technology Lab
Robert-Mayer-Str. 10, 60325 Frankfurt
{hammerla, mehler}@em.uni-frankfurt.de

Abstract

We introduce Gutenberg+, a temporally more faithful version of the Project Gutenberg (PG) corpus, one of the most widely used resources for diachronic text analysis. Despite its popularity, the PG corpus contains a major yet overlooked flaw: around 15% of its entries are collections (e.g., anthologies of books, letters, or poems) rather than atomic works, which distorts temporal analyses since such collections may span multiple decades. We present an automatic method to detect and split these collections into their constituent works, producing a finer-grained and temporally consistent corpus. We further re-annotate publication years using LLM-based retrieval-augmented generative methods, demonstrating the potential of LLMs to enhance structured linguistic resources. To illustrate the utility of Gutenberg+, we conduct a small-scale diachronic case study on negation, showing that our refined corpus captures more nuanced cross-linguistic variation than the original PG data. Finally, we release the corpus in UIMA format with full metadata and linguistic annotations, providing a standardized resource for future research on diachronic language change.

Keywords: Corpus, Digital Libraries, Multilinguality, Text Analytics

1. Introduction

In the current LLM-dominated research landscape, large-scale unstructured corpora are widely used for training and evaluation. However, the reliability of downstream linguistic analyses, such as diachronic text analysis (DTA), critically depends on the structural and metadata integrity of these resources. While scale has increased dramatically, inconsistencies in document structure and coarse-grained temporal metadata can distort empirical findings and compromise reproducibility. Among such resources, the Project Gutenberg (1971-2026) (PG) collection stands out for its scope and accessibility, encompassing thousands of digitized literary works that, while technically spanning more than a millennium of writing, provide substantial and continuous coverage for roughly the past four centuries of language use (Gerlach and Font-Clos, 2020). Although the PG corpus holds great potential, its utility for DTA is constrained by noisy metadata, particularly incomplete or inaccurate publication information. These limitations hinder reliable temporal analysis and prevent the corpus from fully realizing its potential as a historical resource. Recent efforts have sought to address this problem and improve the temporal reliability of the PG corpus. For instance, Hegde et al. (2025) introduced CHRONOBERG, a temporally structured corpus of English book texts spanning 250 years, curated from Project Gutenberg and enriched with external temporal metadata from sources such as OpenLibrary and Wikipedia. Similarly, Momen et al. (2025) tackled the challenge of missing temporal annotations by leveraging LLMs with retrieval-augmented generation to estimate the production years of all PG books, potentially enhancing the corpus's usability for diachronic research. While these endeavors represent significant progress, they share a critical limitation: out of 76,534 items in the PG corpus, 11,650 ($\sim 15\%$) are collections, such as volumes of poems, anthologies, collected works, or books of letters. These collections often span multiple decades, meaning that their temporal annotations do not accurately reflect the underlying texts. To properly study the PG corpus through a diachronic lens, such collections must be split into their constituent items, creating a more fine-grained and temporally faithful resource. We address this structural limitation by transforming PG into a finer-grained, temporally consistent, and richly annotated resource suitable for reliable linguistic analysis and evaluation. As a test case, we quantify the impact of our more fine-grained corpus view through the lens of negation, a core linguistic phenomenon characterized by its dynamic nature, making it particularly suitable for examining language change over time (e.g. Jespersen et al., 2025; Croft, 1991; Mazzon, 2016). Our contributions are threefold:

1. We split the collections in the PG corpus into atomic parts, creating a finer-grained, temporally faithful representation of the texts.

2. We re-annotate the new split using the approach of Momen et al. (2025) and add negation annotations across nine languages using the D-Neg tool (Hammerla et al., 2025), which extends the NegBERT framework (Khandelwal and Sawant, 2020). We make this re-

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 86–92
May 11, 2026.

source publicly available for the research community.

3. We demonstrate the analytical impact of our structural refinement through a multilingual diachronic case study on negation, showing that fine-grained metadata significantly enhances the prediction of temporal distributions.

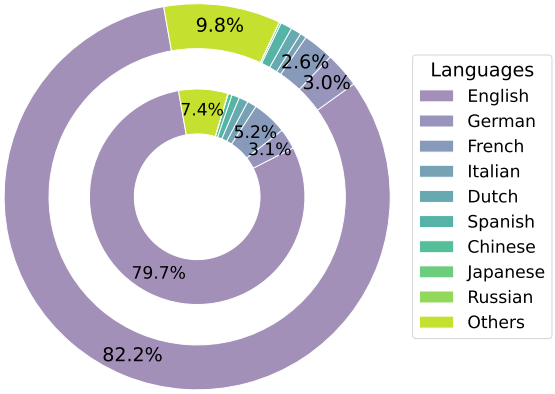


Figure 1: Distribution of items by language in the PG corpus, showing all languages for which negation can be detected, with the remaining languages grouped into the "Others" category. The inner ring illustrates the initial PG split; the outer ring depicts the fine-grained split produced by our version of the PG.

2. Dataset

2.1. Project Gutenberg

Our base corpus is derived from Project Gutenberg (1971-2026), a large-scale digital library of public-domain texts digitized by volunteers. The corpus comprises 76,534 items in 70 languages. English (61,016), French (4,010), Finnish (3,414), German (2,357), and Italian (1,075) are the most frequent languages, together accounting for about 94% of all items. The corpus is further organized into 474 virtual bookshelves. The largest categories are Novels (23,676 items), British Literature (9,758), Adventure (8,238), American Literature (7,584), and Children & Young Adult Reading (6,713). In addition, Project Gutenberg provides subject metadata, containing 41,423 distinct subject annotations. Among the most frequent are the Library of Congress Classification (LCC) labels[1] PS (12,068), PR (10,705), PZ (7,937), PQ (5,327), and PT (3,295).

2.2. Detecting and splitting Collections

To identify and split collections within the PG corpus, we first detect which items constitute collections. We leverage the LLM-generated summaries available in the corpus metadata for this purpose. Manual inspection revealed that PG items whose summaries contained the word collection within the first three sentences were almost always collections. A small-scale human evaluation (100 samples) confirmed this heuristic with 98% precision, which we prioritize to ensure that predicted positives are highly reliable. Any such text collection hinders the application of procedures that operate item-wise, such as publication year annotators. Therefore, mapping the members of these collections to individual items is indispensable.

After identifying the collections, we proceed in two steps. (1) We extract the table of contents of each collection using Algorithm 1, obtaining the titles of the individual items contained within. (2) We then split the collection text into its atomic parts using Algorithm 2. This process involves normalizing the text and detecting lines that directly match an item title surrounded by newline characters, which typically indicate a section heading. To assess the reliability of our collection-splitting procedure, we conducted a small-scale human evaluation on 100 randomly selected collections and achieved an accuracy of 86% for correctly segmented (i.e., satisfying) splits. The two primary sources of error were: (1) collections lacking a clear table of contents, which prevented the identification of distinct items (57% of errors), and (2) cases where item titles were detected from the table of contents but could not be matched to the corresponding sections in the text due to missing or inconsistent headlines (29% of errors). Given the absence of a standardized digitization format in the PG corpus, structural inconsistencies across editions remain an unavoidable limitation.

2.3. Publication Year Annotation

We refine the temporal dimension of the complete PG dataset by reusing the LLM-based retrieval-augmented publication year annotation method proposed by Momen et al. (2025), which has proven to be both effective and efficient. Specifically, we adopt their setup using gemma-2-9b-it with 8-bit quantization and the same prompting configuration. Following their best-performing setting, we provide only the first and last page of each item as context for year prediction. Since most PG items lack explicit page boundaries, we approximate page length by taking the first and last 3,000

[1]The reported LCC labels denote broad literature classes: PS = American literature, PR = English literature, PZ = fiction and juvenile belles lettres, PQ = French, Italian, Spanish, and Portuguese literature, and PT = German, Dutch, and Scandinavian literature.

Input: Collection text T, title t, search window w
Output: Chapters C, remaining text R
1. Split T into lines L; restrict search to first w lines.
2. Find first short line in L containing keyword from KEYWORDS $\rightarrow$ toc_start.
3. If not found and t contains a comma, return $[(t)], T$.
4. From toc_start onward, collect non-empty lines l with $2 < |l| < 40$, remove trailing numbers.
5. Stop after three consecutive empty lines.
6. Remove entries in PAGE_WORDS.
7. Split C into main chapters (ending with ":") and subchapters.
8. Return whichever set is non-empty along with remaining text.

Algorithm 1: Extract Table of Contents

Input: Collection text T, chapter titles C
Output: Items I, filtered chapters C', line ranges R
1. Split T into lines L.
2. For each chapter c_i in C, find candidate start lines using FINDNORMALIZEDLINE.
3. Keep the first valid start appearing after previous ones; record indices of invalid matches.
4. Append end of L to list of start positions.
5. Segment L between successive start positions $\rightarrow$ items I.
6. Remove unmatched chapters; assert $|I| = |C'|$.
7. Return I, C', and line ranges R.

Algorithm 2: Split Collections into Items

characters of each text.

2.4. Segmentation and Negation Annotation

For sentence and word segmentation, we use spaCy (Honnibal et al., 2020), employing its small model family[2]. For the negation detection, we use D-Neg (Hammerla et al., 2025). D-Neg expands the NegBERT architecture (Khandelwal and Sawant, 2020) for multiple languages and refines the classification using syntactic features such as part-of-speech tags and dependency trees, which we also extract with spaCy. The multilingual transformer backbone of D-Neg is EuroBERT (Boizard et al., 2025). The model is implemented as a token-classification pipeline. In a first pass, it identifies negation cues in the input sequence. In a second pass, for each detected cue, it predicts the scope of that cue by classifying each token in the sentence with respect to whether it falls within the corresponding negation scope. This procedure allows us to detect both negation cues and their associated scopes for 9 languages (see Figure 1). The detected cue types include affixal negation, negative adverbs, determiners, pronouns, prepositions, verbal negation cues, and discontinuous cues.

2.5. Corpus Statistics and Release

While the original PG corpus contained 76,534 entries, splitting its 11,650 collections (15%) expanded our Gutenberg+ corpus to 215,960 individual items. After segmentation, our corpus comprises 217.15 million sentences and 5.35 billion tokens, spanning one millennium of writing (see Figure 2). We identify 51.51 million instances of negation across nine languages, covering over 90% of all items in both the original PG corpus and our fine-grained version (see Figure 1). For distribution of our refined corpus we chose UIMA because it provides a standardized framework for representing and exchanging richly annotated text corpora (Ferrucci and Lally, 2004). UIMA facilitates integration with a variety of NLP tools and workflows, making it straightforward for other researchers to load, process, and extend our annotated corpus [3].

3. Experiments and Analysis

We propose two complementary approaches to evaluate the impact of our fine-grained split: comparing item frequency over time between the original and refined corpora, and examining language-specific differences in negation distributions. For all pairwise comparisons between distributions P and Q, we employ the Jensen–Shannon Divergence (JSD), a symmetric and bounded information-theoretic measure that generalizes the Kullback-Leibler divergence and is applicable to both univariate and multivariate, discrete, continuous, and mixed distributions:

$$\mathrm{JSD}(P \parallel Q) = \frac{1}{2}\mathrm{KL}(P \parallel M) + \frac{1}{2}\mathrm{KL}(Q \parallel M),$$

where $M = \frac{1}{2}(P + Q)$, and KL denotes the Kullback-Leibler divergence. To correct for bias in the JSD estimate, we use a bootstrap-based bias correction (Efron and Tibshirani, 1994). Let $\mathrm{JSD}(P \parallel Q)$ be the original JSD estimate from the

[2] `en_core_web_sm`, `de_core_news_sm`, etc., depending on the language. If no language-specific model is available, we fall back to the English model.

[3] Our dataset is publicly available on HuggingFace and can be downloaded from `https://huggingface.co/datasets/PG-S/PG_FG`.

sample data, and let $\overline{\mathrm{JSD}}_B$ be the mean of the bootstrap JSD estimates. The bias is then estimated as:

$$\text{Bias} = \overline{\mathrm{JSD}}_B - \mathrm{JSD}(P \parallel Q).$$

The bias-corrected JSD estimate is:

$$\mathrm{JSD}_{\text{corrected}}(P \parallel Q) = \mathrm{JSD}(P \parallel Q) - \text{Bias}$$

$$= 2 \cdot \mathrm{JSD}(P \parallel Q) - \overline{\mathrm{JSD}}_B.$$

The bias-corrected $100(1-\alpha)\%$ confidence intervals for the JSD are computed as:

$$\mathrm{CI}_{\text{lower, corrected}} = 2 \cdot \mathrm{JSD}(P \parallel Q) - \mathrm{JSD}_{B,1-\alpha/2},$$

$$\mathrm{CI}_{\text{upper, corrected}} = 2 \cdot \mathrm{JSD}(P \parallel Q) - \mathrm{JSD}_{B,\alpha/2},$$

where $\mathrm{JSD}_{B,\alpha/2}$ and $\mathrm{JSD}_{B,1-\alpha/2}$ are the $\alpha/2$-th and $1-\alpha/2$-th percentiles of the bootstrap JSD estimates, respectively.

3.1. Temporal Item Distribution

To assess how the two corpus versions differ temporally, we compare the distributions of yearly item frequencies in the original PG split and our fine-grained PG split, with frequencies normalized by the total number of items in each corpus. In addition to the raw yearly frequencies, we compute LOWESS-smoothed frequency curves (with smoothing parameter frac $= 0.2$) to highlight broader temporal trends (see Figure 2). Both corpora are annotated using the publication-year annotation method of Momen et al. (2025) (see Section 2.3 for details of the experimental setup).

3.1.1. Results and Analysis

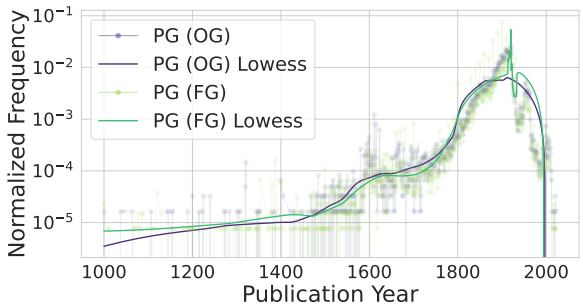


Figure 2: Comparison of yearly item frequencies between the original PG split and our fine-grained PG split, with frequencies normalized by the total number of items in each corpus (OG = original; FG = fine-grained).

From the LOWESS-smoothed distribution of our fine-grained corpus, we observe a more pronounced rise in the number of items leading up to 1920, followed by a sharp decline thereafter, with item counts dropping to roughly 10% of their 1920 level compared to the original PG split. Importantly, divergence decreases monotonically over time, suggesting that structural inconsistencies are concentrated in earlier, sparsely populated corpus sections. We quantify this trend by computing $\mathrm{JSD}_{\text{corrected}}$ over 200-year intervals between 1000 and 2000 (see Table 1). Divergence is highest for early slices (e.g., 1000–1199: 0.587) and lowest for later periods (e.g., 1800–1999: 0.047), with an overall divergence of 0.047. While the aggregate divergence across the full time span is moderate, early periods exhibit substantial divergence (0.59), indicating that coarse collection-level grouping disproportionately distorts sparsely represented historical slices.

Time Slice	JSD	95% CI
1000–1199	0.5867	[0.5067, 0.6800]
1200–1399	0.3798	[0.3058, 0.4508]
1400–1599	0.0654	[0.0422, 0.0910]
1600–1799	0.0574	[0.0495, 0.0667]
1800–1999	0.0467	[0.0455, 0.0480]
full	0.0471	[0.0458, 0.0486]

Table 1: $\mathrm{JSD}_{\text{corrected}}$ and 95% $\mathrm{CI}_{\text{corrected}}$ for 200-Year time slices between the original PG split and our fine-grained split.

3.2. Per-Language Distributional Effects

For the second experiment, we examine whether the fine-grained split captures diachronic linguistic phenomena, such as negation, more distinctly across languages. We include all languages for which D-Neg supports negation detection and that are represented by at least 100 items in the corpus, yielding the following set: English, Italian, French, German, Spanish, Dutch, and Chinese. To this end, for each language we construct a joint empirical distribution over publication year and negation frequency, where negation frequency is computed by first dividing the total number of detected negation cues in an item by the number of sentences in that item and then averaging this value across all items published in the same year. We use the publication-year annotations described in Section 2.3 and the negation annotations described in Section 2.4. We then compute pairwise $\mathrm{JSD}_{\text{corrected}}$ between these language-specific distributions in both corpus splits (see Figure 3). This allows us to assess whether the fine-grained version yields more differentiated and linguistically interpretable cross-linguistic relationships, including similarities and dissimilarities potentially associated with properties such as negative concord or shared language family.

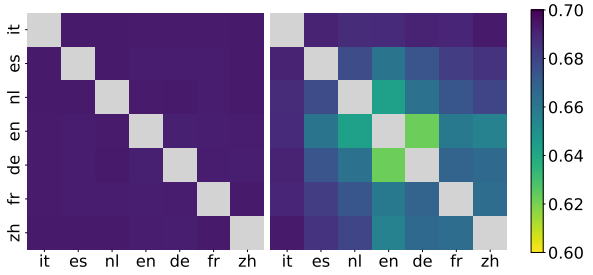


Figure 3: Pairwise $\mathrm{JSD}_{\text{corrected}}$ between language-specific joint distributions over publication year and year-level average negation frequency for the original (left) and fine-grained (right) PG splits.

3.2.1. Results and Analysis

In the original PG split, the language-specific joint distributions of publication year and negation frequency exhibit uniformly high JSD values, suggesting limited differentiation across languages, which may obscure typologically meaningful relationships. In contrast, the fine-grained split reveals more structured variation, including lower divergence among typologically related languages (e.g., Germanic non-concord languages such as English, German, and Dutch). We do not interpret this greater dispersion as inherently better. Rather, we take it to be desirable insofar as it reduces near-uniform divergence patterns that may result from noisy publication-year metadata and yields relationships that are more consistent with independently established typological structure in negation (Haspelmath, 2013). English appears relatively central, with low divergence to most languages, whereas Italian remains more isolated. Italian's more isolated position may partly reflect language-specific negation properties, as Standard Italian has been analyzed as a non-strict negative concord language (Poletto and Oliviéri, 2018), although corpus-compositional factors may also contribute. English, by contrast, may appear more central because its larger corpus representation likely yields a more stable aggregate profile. To quantify dispersion across language pairs, we compute the Interquartile Range (IQR) of the divergence values. The IQR increases from 0.001 in the original split to 0.025 in the fine-grained split, a 25-fold increase, indicating substantially greater cross-linguistic variability. A two-sample Kolmogorov–Smirnov test confirms that the distributions differ significantly ($D = 0.9524$, $p < 0.001$).

4. Related Work

DTA on the PG Corpus. Several studies have applied DTA to the PG corpus. These include investigations of the stylistic evolution of literary authors over time (Klaussner and Vogel, 2018) and their influence (Hughes et al., 2012), the classification of diachronic semantic shifts (Koldenhof, 2021), and detailed analyses of changing emotional or thematic patterns, such as the study of melancholy (Kung, 2007).

PG Publication Year Refinement. To date, only a few studies have attempted to refine the temporal labels of PG texts. Gerlach and Font-Clos (2020) relied on author lifespan estimates, Hegde et al. (2025) augmented these with external sources such as Wikipedia and OpenLibrary, and Momen et al. (2025) combined previous approaches with LLM-based RAG applied to the first and last pages of each item.

5. Conclusion

We address a critical limitation of the widely used Project Gutenberg (PG) corpus that has been largely overlooked: approximately 15% of its entries are collections rather than individual works. This structural issue undermines the validity of diachronic analyses on PG. By automatically detecting and splitting these collections into their atomic items (e.g., books, letters, poems), we produce a more temporally faithful and linguistically coherent resource. Using negation as a case study, we demonstrate that our fine-grained corpus enables more precise and interpretable analyses of diachronic linguistic phenomena. To support further research, we publicly release the fully formatted corpus in UIMA format, including all metadata, publication year annotations, segmentation, and negation annotations. Our work establishes a stronger foundation for large-scale historical language studies and underscores the importance of structural and metadata integrity in computational linguistics.

6. Limitations & Future Work

For publication-year annotation, we adopt the method of Momen et al. (2025) and use the same models as in their original setup. These models are compact and efficient, which is beneficial for a corpus at the scale of Project Gutenberg, although future work could examine whether larger models yield more accurate temporal annotations. Our approach to splitting collections currently relies on an effective heuristic; future work could explore whether this step can be handled more systematically with LLM-based methods. Finally, our analysis focuses on negation. Extending the evaluation to additional linguistic phenomena would help determine how broadly the observed effects generalize.

7. Ethics Statement

Our work builds on the Project Gutenberg corpus, a collection of public-domain texts, and does not involve any sensitive, private, or personal data. All texts used are freely available and legally redistributable, and our annotations and processed corpus are released in compliance with PG's terms of use.

We acknowledge that the PG corpus is biased toward Western, English-language, and literary texts, thus underrepresenting other languages, genres, or historical perspectives. Our fine-grained corpus does not mitigate these intrinsic corpus biases; users of Gutenberg+ should remain aware of these limitations when drawing linguistic or cultural conclusions.

8. Acknowledgements

This work was supported by the German Research Foundation (DFG) as part of Project INF within CRC 1629 "Negation in Language and Beyond" (NegLaB - https://www.neglab.de/).

9. Bibliographical References

Nicolas Boizard, Hippolyte Gisserot-Boukhlef, Duarte M. Alves, André F T Martins, Ayoub Hammal, Caio Corro, Céline Hudelot, Emmanuel Malherbe, Etienne Malaboeuf, Fanny Jourdan, Gabriel Hautreux, João Alves, Kevin El-Haddad, Manuel Faysse, Maxime Peyrard, Nuno M Guerreiro, Patrick Fernandes, Ricardo Rei, and Pierre Colombo. 2025. EuroBERT: Scaling Multilingual Encoders for European Languages. In *Second Conference on Language Modeling*, pages 1–28, Montreal, Canada.

William Croft. 1991. The evolution of negation. *Journal of Linguistics*, 27(1):1–27.

Bradley Efron and R.J. Tibshirani. 1994. *An Introduction to the Bootstrap*. Chapman and Hall/CRC.

David Ferrucci and Adam Lally. 2004. Uima: an architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10(3–4):327–348.

Martin Gerlach and Francesc Font-Clos. 2020. A standardized project gutenberg corpus for statistical analysis of natural language and quantitative linguistics. *Entropy*, 22(1).

Leon Lukas Hammerla, Andy Lücking, Carolin Reinert, and Alexander Mehler. 2025. D-neg: Syntax-aware graph reasoning for negation detection. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pages 1432–1454, Mumbai, India. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics.

Martin Haspelmath. 2013. Negative indefinite pronouns and predicate negation (v2020.4). In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Zenodo.

Niharika Hegde, Subarnaduti Paul, Lars Joel-Frey, Manuel Brack, Kristian Kersting, Martin Mundt, and Patrick Schramowski. 2025. Chronoberg: Capturing language evolution and temporal awareness in foundation models.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

James M. Hughes, Nicholas J. Foti, David C. Krakauer, and Daniel N. Rockmore. 2012. Quantitative patterns of stylistic influence in the evolution of literature. *Proceedings of the National Academy of Sciences*, 109(20):7682–7686.

Otto Jespersen, Brett Reynolds, Peter Evans, and Olli O. Silvennoinen. 2025. *Negation in English and other languages*. Number 8 in Classics in Linguistics. Language Science Press, Berlin.

Aditya Khandelwal and Suraj Sawant. 2020. NegBERT: A transfer learning approach for negation detection and scope resolution. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5739–5748, Marseille, France. European Language Resources Association.

Carmen Klaussner and Carl Vogel. 2018. A diachronic corpus for literary style analysis. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Dylan Koldenhof. 2021. Word embeddings to classify types of diachronic semantic shift. University of Twente.

Sally Kung. 2007. A diachronic study of melancholy in a british novel corpus. University of Birmingham.

Gabriella Mazzon. 2016. *A History of English Negation*. Routledge.

Omar Momen, Manuel Schaaf, and Alexander Mehler. 2025. Filling the temporal void: Recovering missing publication years in the Project Gutenberg corpus using LLMs. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 17318–17334, Vienna, Austria. Association for Computational Linguistics.

Cecilia Poletto and Michèle Oliviéri. 2018. *Chapter 9. Negation patterns across dialects*, pages 133–148. John Benjamins Publishing Company.

10. Language Resource References

Project Gutenberg. 1971-2026. *Project Gutenberg Corpus*. Project Gutenberg. Open-access digital library; available at https://www.gutenberg.org, accessed August 22, 2025.

Automatic Lemmatisation for Norwegian

Ahmet Yıldırım, Kristin Hagen, Dag Trygve Truslew Haug
Humit, Faculty of Humanities, University of Oslo
{ahmetyi,kristiha,daghaug}@uio.no

Abstract

We report on a new lemmatisation system for Norwegian, which is a particularly challenging language with two written standards, Bokmål and Nynorsk, that both have a lot of optionality. Our system covers both varieties and consists of a neural model that classifies words into rewrite rule classes that produce their lemma, as well as a large-scale computational lexicon of Norwegian that gives all possible inflections of a large part of the Norwegian vocabulary. We test different ways of combining these components. When evaluated with pure string-matching against the lemmas in the gold data, all systems perform approximately at the same level (99.1-99.2% on Bokmål and 98.5-98.6% on Nynorsk), but detailed error analysis shows that the computational lexicon reduces the number of true errors by more than half (reaching 99.6% accuracy on Bokmål and 99.3% on Nynorsk), as opposed to "surface errors" like using a different, but equally acceptable spelling variant of the correct lemma.

Keywords: lemmatisation, BERT models, computational lexicons, Norwegian

1. Introduction

Lemmatisation is the task of reducing variable forms of a word into a base form, typically the one used as the dictionary entry of the word. It is a task that has become somewhat less popular in recent years. It last figured in the SIGMORPHON shared tasks in 2019 (McCarthy et al., 2019). In later years, these tasks have focused on morphological generation, where lemmas are typically part of the input data rather than predicted. Within the natural language processing field, lemmatisation as a way of addressing the sparse data problem by eliminating morphological variants has largely given way to the use of embeddings. But for many text analysis tasks in the humanities, such as concept analysis or corpus-assisted lexicography, lemmatisation is clearly still very useful, precisely because it serves to group different word forms that are relevant to one and the same concept, or one and the same dictionary entry.

In this paper, we report on the development of a lemmatisation system for Norwegian. Norwegian is relatively well equipped with lexical resources, but nevertheless presents some unique challenges for lemmatisation related mainly to the fact that the language has two written varieties, Nynorsk and Bokmål, which are close enough to be sometimes indistinguishable and which both allow a considerable amount of spelling variation, including in lemma forms. Lemmatisation systems for Norwegian, including the one we present here, must cover both varieties and all the variation within them. Careful error analysis is therefore important to correctly evaluate the quality of such systems, to distinguish between substantial errors, where the system has picked the wrong lemma, or produced a nonexistent lemma, from those where the system has merely chosen a different way of spelling the correct lemma.

We test several different setups: a state-of-the-art system for neural multilingual lemmatisation, and two hybrid systems where the neural system intersects with a large-scale computational lexicon in two different ways. While the hybrid systems are language-specific and require resources that are not available for most languages, their performance in comparison with purely neural systems also tells us something about the strengths and weaknesses of the latter systems.

In Section 2 we detail some of the specific challenges in lemmatising Norwegian text. Then in Section 3 we survey related work in lemmatisation. Section 4 describes our different experiment setups and their components, while Section 5 evaluates their performance and provides detailed error analysis. Section 6 provides details on the implementation of the lemmatiser inside the overall Humit tagger and Section 7 concludes.

2. Lemmatisation for Norwegian

For historical reasons, there exist two different written standards for Norwegian, called *Bokmål* and *Nynorsk*. Bokmål is the majority standard, used by around 90% of the population, while Nynorsk is used by around 10% of the population, mainly in the western parts of the country (Karterud et al., 2024). The two varieties are close enough that sentences, especially short ones, may be indeterminate as to whether they are in Bokmål or Nynorsk. Many lemma forms are the same, but there are also lemmas that differ. Moreover, as language purism has been relatively strong within Nynorsk, there are a number of words, in particular of Low German origin, that are not considered proper in Nynorsk. Many of these words are rela-

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 93–103

tively common in the dialects and are, in fact, found in Nynorsk texts even if not officially acceptable. Furthermore, as the two varieties coexist within the same public discourse, it is relatively common to find e.g., quotes in Bokmål within an otherwise Nynorsk text and vice versa.

When lemmatising, we clearly want to use the standard that the text is written in, whether it is Bokmål or Nynorsk. But some practical questions arise: when we meet stray Bokmål words in Nynorsk text, we may want to deal with these words as proper Nynorsk words even if they officially do not exist. On the other hand, when we encounter longer stretches (often quotes) of Bokmål in Nynorsk text, we may want to deal with them in the same way as when we encounter English, German, or other foreign languages. This is, for example, what the Norwegian Dependency Treebank (Solberg et al. 2014; National Library of Norway 2014) does. In such cases, they apply the postag `unknown` and use the surface form as the lemma. This is the same policy as applied to e.g. English when it occurs in quotes and titles, but when it happens with Bokmål inside Nynorsk, it is confusing from a machine learning perspective that most of the words that are tagged in this way in fact also exist in Nynorsk. For example, in the sentence *Jeg har alltid lurt på hvem som vasker for vaskehjelpen.* ('I have always wondered who washes for the housekeeper'), there are four words that give this away as Bokmål; as a result, when this sentence is quoted inside a Nynorsk text, all words including the six neutral ones, receive the `unknown` tag and are lemmatised by their surface form, although in other parts of the corpus, where they occur in a Nynorsk sentence, they receive other tags and lemmas.

In addition to the complexity of having two written standards of Norwegian, both standards also allow for much more internal variation than what we typically find in other languages. Most of this variation is found in the morphology, but crucially, some of it affects the way the lemma is spelled. For example, infinitives in Nynorsk may end in *-e* or *-a*, and many verbs have an optional infixed *-j-*. This means that the infinitive 'to think' can be spelled *tenka, tenke, tenkja, tenkje*. As another example, the adverb 'forward' in Bokmål, which is very common as a prefix in Norwegian, can be spelled *fram* or *frem*. In such cases, it is not entirely clear what form we should target: one approach would be to use the form of the text we lemmatise, but that may not be clear, or the text may be inconsistent. Moreover, using the standard of the text that is being lemmatised makes it harder to do comparisons *across* texts. Finally, from a practical point of view, spelling variation of this type affects the evaluation of lemmatisation models, which is typically done by string matching.

As a final complication, we mention that Norwegian has undergone several spelling reforms in modern times, the most recent ones in 2005 (Bokmål) and 2012 (Nynorsk). For example, infinitives ending in *-a* were allowed for some but not all verbs in Bokmål until 2005. This affects lemmatisation and morphological analysis of historical texts.

3. Related Work

Automated lemmatisation of Norwegian text goes back at least to Fjeldvig and Golden (1984). This was an early attempt at so-called root lemmatisation, i.e. to link tokens to their root, ignoring, e.g. part of speech and derivational morphology, rather than to actual lemmas/dictionary entries. The system implemented rules that aimed to strip common prefixes and suffixes from Norwegian words, but without word-specific information as would be found in a lexicon. Subsequent work throughout the 1990s focused on developing a full computational lexicon of Norwegian, which eventually resulted in *Norsk ordbank* (Norwegian word bank, National Library of Norway 2022a,b). This resource is currently maintained by the Norwegian language council in collaboration with the University of Bergen. As mentioned above, there were several spelling reforms during this period. As a result, the paradigms in Norsk ordbank had to be marked with a `from_date` and a `to_date` attribute, indicating during which period a particular paradigm was valid. Handling this in a lemmatisation system is not entirely straightforward. On the one hand, one may want the system to be able to handle spellings that are no longer part of the norm. On the other hand, it is not desirable that the system produce forms that are not part of the norm as lemmas for forms that also has an alternative lemma that is part of the norm.

Norsk ordbank can be used to generate all possible inflected forms of the lemmas it contains, and their associated morphological features. Such a resource, in turn, can be used to generate all possible analyses (lemma + morphological features) of words in running text. However, it does not help with picking the correct analysis in context. To deal with this, a system of Constraint Grammar-based rules was developed called the Oslo-Bergen Tagger (Johannessen and Hagen, 2003). In later work (Johannessen et al., 2012), a statistical disambiguating system called OBT+stat was added to the rule-based system to deal with cases where the rule-based system accepted two or more alternative analyses. For lemmatisation, OBT+stat picked the globally most frequent lemma alternative in a large Norwegian web corpus (NoWaC, Guevara 2010). The reported accuracy for lemma-

tisation was 98.48% for Bokmål. The system did not handle Nynorsk.

Haug et al. (2023) showed that a neural architecture based on the NB-BERT model (Kummervold et al., 2021) on its own outperformed both the rule-based system for morphological tagging and a combined system using both the rules and a neural network. They suggested that rules may still be useful for lemmatisation, but left that for future research.

In the meantime, NorBench (Samuel et al., 2023) implemented the general UD lemmatisation pipeline as introduced by Straka et al. (2019) as part of a system to benchmark Norwegian language models. While the focus of that work is not to improve lemmatisation for Norwegian, but to assess language models' understanding of (among other things) Norwegian morphology, their experiments do provide useful benchmarks for purely neural lemmatisation of Norwegian. They report a 99.1% accuracy for lemmatisation on the Norwegian UD treebanks; however, they do not release the fine-tuned models. Our work tries to improve performance by combining this approach with the lexicon from *Norsk ordbank* by fine-tuning BERT models on the same dataset, the Norwegian Dependency Treebank.

Constructing hybrid systems for lemmatisation has been tried before, e.g. by Milintsevich and Sirts (2021), who started from a sequence-to-sequence model for lemmatisation (the Stanford neural lemmatisation system, Qi et al. 2020). However, we observe that on Danish and Swedish, which are closely related to Norwegian, this system generally performs worse than the system of Straka et al. (2019). We therefore chose to start from the latter.

4. Experimental Setup

The goal is to develop better lemmatisation capacities for Norwegian. To do this, we train a lemmatisation model by using the Norwegian Dependency Treebank dataset. Two of the configurations we test in this paper additionally use a morphological classifier as input for lemmatisation. Therefore, we also train a morphological classifier. We want to preserve the traditional tagset for Norwegian that has been used since the Oslo-Bergen Tagger in the nineties. We therefore target the tags of the original Norwegian Dependency Treebank rather than the converted UD ones (Øvrelid and Hohle 2016; National Library of Norway 2023), which means we train on the original treebank. The tagset covers both part-of-speech tags and a more detailed morphosyntactic description, including person, number, gender etc. Only the part-of-speech tag is used in the lemmatiser, but this lemmatiser is embedded in a larger system that also produces a full morphosyntactic description.

split	bokmål	nynorsk	total
train	280369	272355	552724
test	30650	30712	61362

Table 1: Data split word counts

Unlike the UD version, this treebank does not have an official train/test split, and so we create such a split with a 0.9/0.1 ratio, respectively.[1] Word counts for this split are given in Table 1. Note that this main train split is later subdivided into task-specific train and dev sets, whereas all evaluation is done on the main test set, to avoid data leaks between tasks, see below in Section 4.3.

Our previous experience with lemmatisation using neural models showed that even with the best models, which achieve very high overall accuracies, the occasional erroneous lemmas obtained can seem very strange for the human eye, for example, by violating hard phonotactic constraints in the language. Therefore, we experiment with different configurations of a single lemmatisation model, plus integration of a morphological classifier and a morphological lexicon. In the following, we give details of UDpipe lemmatisation pipeline, the lexicon, the machine learning setup, and the different experimental configurations.

4.1. The UDpipe Lemmatisation Pipeline

The idea of the UDpipe lemmatisation pipeline (Straka et al., 2019) is to find a rule that edits the word to get the lemma. In the first phase, the system looks at all word-lemma pairs and extracts rules, viz. edits that will rewrite the word as the lemma. Then, a model is trained that classifies input tokens into rule classes in a multi-class classification setting. In the inference phase, when a word and its rule are given, the UD lemmatisation pipeline applies the rule to the word to get the lemma.

The rules of the pipeline are created in a very general way to allow processing of as many languages as possible. A rule has four features: casing, prefix operations, suffix operations, and whether it represents a whole replacement with an absolute string lemma. The operations are copying/deletion/addition of characters in specific locations of the word with reference to the beginning (prefix) and end of the word (suffix). However, we

[1] This split is published in the `ndt_1.0_nob` and `ndt_1.0_nno` directories of https://github.com/humit-oslo/humit-tagger-sources, where all the training data for the Humit tagger system are available, including data for tasks not discussed in this article.

form	language	POS	lemma
arbeider	nno	verb	arbeide
arbeider	nob	noun	arbeid
arbeider	nob	noun	arbeide
arbeider	nob	noun	arbeider
arbeider	nob	verb	arbeide

Table 2: Sample lexicon entries. nno: Nynorsk, nob: Bokmål.

modified the code implemented by Samuel et al. (2023)[2], to make the rule extraction prefer suffix operations by inserting constant dummy characters at the beginning of both lemma and form. This was motivated by most inflections in Norwegian being suffixing, which also simplifies the classification tasks. To give an example for a rule, the word "planlagt" and the lemma rule "lower;-+e→-+ge" produce the lemma "planlegge". Here, the first part of the lemma rule indicates that the word will be lower-cased, and the second part indicates the following: add *e* at the end, replace the second last character with *g*, and replace the fourth last character with *e*.[3]

4.2. The Morphological Lexicon

The morphological lexicon we use is *Norsk ordbank*, mentioned above. From this lexicon, we extracted all possible forms, along with the information about language form (Bokmål, Nynorsk, or both), the lemma, and part of speech, yielding quadruples as in Table 2, which shows different forms from the root *arbeid* 'work'.

Norsk ordbank is a carefully hand-crafted resource that has been developed over a long period. This means that the analyses found there are, in principle, valid. Nevertheless, there are certain complications. First, we use part-of-speech to disambiguate; for example, in Table 2, if we know that the form *arbeider* is a verb, only the lemma *arbeide* is possible. However, the morphological analysis could be wrong. This can happen simply because the tagger makes an error, or because of discrepancies in the tags between the tagger and the lexicon. Such discrepancies arise because the part-of-speech classification that is used in the Norwegian Dependency Treebank differs in some details from the classification in *Norsk ordbank*. For example, the Norwegian Dependency Treebank treats words of location such as *her* 'here', *hjemme* 'at home' etc. as prepositions, while the lexicon uses the traditional classification of such words as adverbs. The differences are small, however, and we leave an eventual unification of the tagsets to future work.

Second, as we saw above, the lexicon contains word forms and inflections that are no longer considered correct. We include these to be able to deal with older texts and with non-standard language. This can sometimes lead to unwanted ambiguities. These could possibly be reduced by implementing a prioritization (only accept a non-standard analysis if no standard analysis exists), but we did not attempt such a strategy.

Third, Nynorsk often contains words that are considered Bokmål and which are therefore not listed as possible Nynorsk words. Here, we do implement a prioritization scheme: We enrich the vocabulary in Nynorsk with all Bokmål forms that do not otherwise occur in the lexicon. We do not similarly enrich the Bokmål lexicon with Nynorsk forms, because the amount of Nynorsk found in Bokmål texts is minuscule.

Fourth, many word forms do not occur in the lexicon at all. One reason is that compounding is a productive process in Norwegian. Recent words may also not yet be included. In these cases, the lexicon obviously does not give us a lemma. Similarly, in cases where the lexicon has more than one analysis, such as for *arbeider* as a noun in Table 2, the lexicon does not give us an analysis on its own.

4.3. Machine Learning Setup

To evaluate models during training, from the main training set, we create task-specific train and dev sets for individual tasks, lemmatisation, and morphological classification, each close to the size of the test set. In this way, while we keep the test set the same across these tasks, we create task-specific train and dev sets to balance class distributions and improve class coverage in the task-specific training processes.

The lemmatisation classifier is trained based on $\text{NorBert}_{3,\text{large}}$ since finetuning this model was reported to be the best performing model for this task (Samuel et al., 2023). Our modified rule extractor extracted 598 rules from the task-specific train set. We use the NorbertForTokenClassification module[4] to train this model. We treat a whole rule as one class, regardless of the rule components, such as casing, absolute rewriting of the lemma, or suffix.

Since the token classification task classifies tokens, the words with multiple tokens are handled as follows: The first token is assigned to the actual class, and the rest of the tokens to a default

[2]The repository for the code is: https://github.com/hplt-project/HPLT-WP4/blob/main/evaluation/ud/lemma_rule.py

[3]See Straka et al. (2019) for an explanation of the formalism.

[4]https://huggingface.co/ltg/norbert3-large/blob/main/modeling_norbert.py

class created for this purpose. This process also identifies the boundaries of words, which the tagger later uses to create its output. Therefore, the loss is computed for all tokens, but not specific tokens such as the beginning or the end token of the word.

Our initial experiments with the hyperparameters given by Samuel et al. (2023) performed poorly compared to the performance measure of 99.1% accuracy reported in their article. We believe this may be due to two reasons: 1) the report is based on a model trained for multiple UD tasks, having multiple classification heads, which leads the base model weights to be updated by training for multiple tasks, and 2) although the source of the datasets are the same, the training and test datasets in our experiments and the ones reported on in the article are different. Therefore, we experimented with several hyperparameters to achieve an accuracy comparable to the reported value, although the two models are not directly comparable.

Based on our experiences with the training process, we settled on using the following hyperparameters: batch size=8, number of epochs=50, and initial learning rate=0.00005. Other parameters are default parameters set by the HuggingFace Transformers Trainer library version 4.41. Throughout the training, we evaluate the model every 500 steps and select the best-performing model according to the accuracy score of the task-specific dev set. In other words, since we are looking for the best model, we update the weights in small batches with a low learning rate and evaluate the model at short intervals. Our lemmatisation model has a classification accuracy of 99.22% for the dev and 99.16% for the test set, which are close to the best previously reported (Samuel et al., 2023).

The morphological classifier model is trained with the same parameters for 267 morphological classes, where each class represents a set of morphological tags. We follow the process detailed in (Haug et al., 2023). We extract each morphological tag combination (such as [subst, prop]) from the training set, and consider each such combination as a different class that a token will be classified into. In line with recent work training and fine-tuning LLMs for Norwegian (Kummervold et al., 2021; Samuel et al., 2023), we train both the lemmatisation classifier and the morphological classifier on datasets consisting of both Bokmål and Nynorsk data. This is especially suitable because the base model that we are fine-tuning was also trained on both written varieties. In this training, the morphological classifier obtained by picking the best dev set classification accuracy has accuracy scores of 98.41% for the task-specific dev set and 98.27% for the test set. The upper bound of the trained model is given by the number of words in the test set that require labels not seen during training. This number is quite small: only 27 words, with one of these occurring twice for a total of 28 occurrences, i.e. less than 0.05%.

4.4. Different Experiment Configurations

We set up three different configurations to experiment with: `model_only`, `model_lexicon`, and `model_lexicon_disamb`.

`model_only` uses only the lemmatisation model. This configuration identifies the lemma rule class of each word and applies the rule to the word to get the lemma.

`model_lexicon` identifies the morphological class of words using the morphological classifier model. If the word is tagged `prop`, then it is considered a proper noun and the lemma is returned as the word itself, retaining upper case. If a proper noun is also tagged as `gen` and there is *s* at the end of the word, then it is considered in genitive form, and the lemma is produced by removing the last *s*. Otherwise, the `model_lexicon` checks the word in the morphological lexicon introduced in Section 4.2, and uses the lemma in this list. If a word (or a compound word) does not exist in the lexicon or if there are multiple suggestions from the lexicon, then the lemma is identified by applying the rule suggested by the lemmatisation model to the word, as in the `model_only` configuration.

`model_lexicon_disamb` applies the same steps as the `model_lexicon` configuration, but it tries to disambiguate among the lemmas if there are multiple suggestions for the word in the lexicon. This configuration, given a word, sorts the rule classes according to their probabilities retrieved from the lemmatisation model and selects the most probable one that produces the lemma among the suggestions of the morphological lexicon. If a word (or a compound word) does not exist in the lexicon, or if the lemmas obtained from the lemmatisation model and the lexicon do not share at least one lemma, then the first suggestion of the lemmatisation model is used, similar to the `model_only` configuration. We use the lemmatisation model's first suggestion to identify word boundaries for all configurations.

5. Evaluation and Discussion

This section evaluates and discusses the performance of the different configurations on the test set. Table 3 shows the number of errors and the accuracy rate of the three configurations on the Bokmål and Nynorsk data. We observe that Nynorsk is slightly harder than Bokmål, likely due to hav-

ing a more complex morphology. On the other hand, the differences between the various setups are small and look insignificant. However, as we will now see, the similarity in the number of errors hides large differences in the kind of errors that are made.

For the detailed error analysis, we focus on `model_only` and `model_lexicon_disamb`. Cursory investigation of the errors from `model_lexicon` showed that it is intermediate between `model_only` and `model_lexicon_disamb` in the types of errors that it makes.

Table 4 shows our categorization of the cases where there is a string mismatch between the gold lemma and the lemma produced by the setups `model_only` and `model_lexicon_disamb`. In the first group, we see real, indisputable errors. We have sorted these into the following categories: *Spelling errors* are errors caused by there being a spelling mistake in the source text. In the gold data these are lemmatised under the correctly spelled lemma. The lexicon obviously cannot handle such cases, as it does not contain misspellings. In principle, the trained model could be able to deal with them by finding a rule that rewrites them into their correct lemma. But since spelling errors are rare and unsystematic, we find small differences between `model_only` and `model_lexicon_disamb` in this category, at least for Bokmål. For Nynorsk, it does seem that the model is able to deal with some spelling mistakes. The second category is *wrong variety*, i.e. cases where the system produces a Bokmål lemma for Nynorsk text or vice versa. The trained model sometimes does this because it has been trained on Bokmål and Nynorsk jointly, as explained above. The third group is *capitalization*: these are due to the morphological tagger wrongly identifying a word as a proper noun (or vice versa). The fourth group is *abbreviations*: we see an interesting difference in how `model_lexicon_disamb` handles these worse in Bokmål. This is because the Bokmål test set contains abbreviations of political parties (*Ap.* for *Arbeiderpartiet*) spelled with a full stop; as these are proper nouns, `model_lexicon_disamb` uses the form as lemma, but the gold data in this case has the abbreviation without a stop. The fifth group is *other errors*. There are errors that cannot be further classified based only the surface forms, but require inspection into how the neural system and the rules interact. They will be discussed in more detail in Section 5.1. But we notice already that other errors are much less frequent with `model_lexicon_disamb` than with `model_only`.

However, there are many mismatches with the gold data which are arguably not real errors. The clearest cases are those where the gold data is simply wrong, which we call *gold errors*. As expected, these are about equally frequent in both configurations, although small differences arise if one configuration makes the same mistake as the gold data. The next group are *optional* forms: as explained in the introduction, many Norwegian lemmas have several equally acceptable spellings. The most common case here is actually the word *TV/tv* 'television' and its compounds, which can be spelled in capitals or in small letters. Finally, we come to the group which makes the biggest difference between `model_only` and `model_lexicon_disamb`, namely *adj/verb*. This is a classical problem in part-of-speech tagging for Norwegian (and other Germanic languages): When is a word form like *spist* 'eaten' an adjective and when is it a verb in the perfect participle form? This also translates into a lemmatisation problem because the morphological lexicon lemmatises the perfect participle as *spise* ('eat', the infinitive of the verb), but the adjective as *spist* ('eaten'). The Norwegian Dependency Treebank, which is the source of our training and test material, follows another standard by lemmatising also the adjective under the infinitive of the verb. For such cases, then, the morphological tagger has learned to categorize them as an adjective, but when this part of speech is used to look up the morphological lemma, we do not find the verbal lemma, only the adjective, and so `model_lexicon_disamb` makes a "mistake". However, this is merely a different convention.

The final group in Table 4 is *"Foreign" text*. Most of these are cases where a longer stretch of Bokmål is quoted in otherwise Nynorsk text, as discussed in Section 2. Swedish sometimes also causes problems because there are many shared words.[5] It would be beneficial if a standard could be established on how to handle such "code-switching" within Norwegian, as it does not seem correct to treat them as completely foreign words. We leave this for future work.

In sum, gold errors, optional variants, and adjective/verb errors are not true errors. However, they make up 54.7% and 53.2% of the errors made by `model_lexicon_disamb` in Bokmål and Nynorsk, respectively. For `model_only`, they only make up 28.1% and 21.3% of the errors. In other words, while the number of errors remains relatively constant between `model_only` and `model_lexicon_disamb`, the number of

[5] English, German, and other languages also occur in NDT as part of quotes or titles, but in this case, the forms are generally not existing Norwegian words, and so the model lemmatises them using the form as the lemma.

	Bokmål		Nynorsk	
Configuration	**errors**	**accuracy**	**errors**	**accuracy**
`model_only`	285	99.1%	456	98.5%
`model_lexicon`	240	99.2%	437	98.6%
`model_lexicon_disamb`	236	99.2%	447	98.5%

Table 3: Number of errors and accuracies on the test set

	Bokmål		Nynorsk	
Error type	**`model_only`**	**`model_lexicon_disamb`**	**`model_only`**	**`model_lexicon_disamb`**
Spelling error	19	22	13	27
Wrong variety	5	1	37	2
Capitalization	19	17	60	37
Abbreviation	3	15	2	2
Other errors	155	44	185	86
Gold errors	29	36	32	38
Optional	20	31	26	48
Adj/verb	31	67	39	152
“Foreign” text	4	3	62	62

Table 4: Error types

true errors is significantly reduced. This is shown in Table 5, which we believe gives a better measure of the performance improvement of `model_lexicon_disamb` over `model_only` than what we see in Table 3.

5.1. Further analysis of “other errors”

Table 4 already contains some analysis of the true errors that the systems make, based on directly observable properties of the test set or the produced forms. For Bokmål these are relatively constant across `model_only` and `model_lexicon_disamb`. For Nynorsk we see that using the lexicon improves capitalization errors and the production of forms of the wrong variety (i.e. Bokmål), although it also hurts the treatment of spelling errors. Nevertheless, for both varieties, the largest reduction comes in the category “other errors” in Table 4. We therefore proceed to a closer analysis of these errors.

Notice that it is hard to say what is going on in this category in the `model_only` configuration when there are no detectable properties of the surface forms like spelling errors or capitalization. The system simply picked the wrong lemma rewriting rule, and the usual difficulties in interpreting the BERT model makes it hard to say why. We therefore focus on the `model_lexicon_disamb` configuration.

Table 6 breaks down the “other errors” of Table 4 depending on whether they are due to errors in the lexicon or to classification errors by the neural model, which may arise whenever a POS-tagging error prevents successful lookup in the lexicon, or because there is no entry in the lexicon, or because there are several entries in the lexicon and the model needs to choose.[6]

We see that the true errors classified as “other errors” largely stem from the trained model. The morphological lexicon is a hand-crafted resource that has been maintained over a long period and does not contain many errors. However, our decision to include forms from the lexicon that are no longer acceptable sometimes induces an error whenever the lemma itself (and not just inflected forms) has had a variant spelling. For example, the Bokmål adjective *grønn* ‘green’ used to have a variant spelling *grøn* in its base form. As mentioned above, we include such forms to be able to deal with non-standard and historical texts, but this means that the neuter form, which has always only been *grønt*, can be lemmatised *grøn* or *grønn*, where only the latter is correct today.

Such cases are rare, and most errors are due to the neural model. A large part of these is due to part-of-speech errors. Such tagging mismatches may lead the system to dismiss a lexicon entry that it should have used. It should be noted that not all such tagging mismatches are true errors; as mentioned above, some are merely differences in the part-of-speech analysis between the lexicon and the annotation in the Norewgian Dependency Treebank.

When we look at the errors that the model makes, many of them are understandable. A common example is the weak/strong inflection ambiguity. Norwegian masculine nouns mark definiteness with a suffix *-en* as in e.g. *gutten* ‘the boy’ or

[6] Two errors that were due to an error in the code are not included here. The error has been corrected in the published implementation.

	Bokmål		Nynorsk	
Configuration	**errors**	**accuracy**	**errors**	**accuracy**
`model_only`	205	99.3%	359	98.8%
`model_lexicon_disamb`	102	99.6%	216	99.3%

Table 5: Errors on the test set, discounting gold errors, optional variants and adj/verb errors

Error type	**Bokmål**	**Nynorsk**
lexicon error	7	4
POS error	18	36
no lexicon entry	10	16
several lexicon entries	7	30

Table 6: Source of "other" errors in `model_lexicon_disamb`

hanen 'the rooster'. In some of these nouns (the so-called strong class), the lemma ends in a consonant: *gutt*. In others (the so-called weak class), the lemma ends in *-e*: *hane*. There are also ambiguous nouns: *listen* may be the definite form of either *liste* ('list') or *list* ('trim'). Similar facts hold for neuters with *-et* in the definite form. It is therefore perhaps not surprising that the model makes errors in adding a spurious *-e* to the lemma, or leaving out a required one. Ideally, the model would be able to generalise over common suffixes, but this is not always the case. For example, *beskyttelse* 'protection' contains the common deverbal suffix *-else*, while no Norwegian noun ends in *-els*. Still, `model_only` lemmatises *beskyttelsen* as *beskyttels*.

We should also point out that `model_only` sometimes makes serious mistakes that violate Norwegian phonotactic rules in such a way as to undermine a human's confidence in the lemmatisation. This is due to the rewrite rules not taking into account what letter is being changed, only the position where something changed. To use an English example, the rule that takes the word *women* to its lemma *woman* can be paraphrased as "change the last but one letter to *a*", and not as "change an *e* in the second last position to an *e*". When such a rule is applied wrongly, it can, depending on the phonological environment, yield very strange results. For example, there are quite a number of Norwegian irregular verbs that display a vowel alternation where an *a* in the past tense corresponds to an *e* in the infinitive lemma, e.g. *hang–henge* 'hang', *hjalp–hjelpe* 'help', *rakk–rekke* 'reach' and many more. From this, the system extracts the rule that the third letter from the end changes to an *e*, and then an additional *e* is inserted at the end. But when the system applies this rule to the verb form *strøk* 'stroked', the result is *steøke*, which contains an impossible vowel sequence *-eø-* that stands out in human eyes.

The overall number of errors is quite small, however, and it is useful to look at the cases where the lexicon does not at all provide an analysis to get an impression of how the model performs. These can be completely new words that have not yet been added to the lexicon, or words formed via compounding, which is a very productive process in Norwegian.

In total, there are 1785 occurrences of words in the Bokmål testset (of total 30650 words) that are not in the lexicon. For 1169 of these (65.5%), typically proper nouns and invariable words, the word form is the same as the lemma. For the Nynorsk test set (total 30712 words), there are 1872 that are not in the lexicon. 1063 of these (56.8%) have a lemma that is identical with the word form. In other words, defaulting to using the form as lemma whenever the word is not in the lexicon would yield a baseline performance of 57-66% on out-of-vocabulary items. However, as expected, the neural system performs much better than that, with an accuracy of 97.4% on these items in Bokmål and 95.8% in Nynorsk. This is, however, noticeably worse than the performance of `model_only` on the whole dataset, indicating that the words that are not in the lexicon are also harder for the neural system, likely because they are rare or nonexistent in the training data.

6. Implementation

We have incorporated the lemmatiser into our Norwegian morphological tagger, which is published in our repository [7]. This tagger prioritizes speed and ease of use, offering more features than the specific experiments described in this paper. While the reported experiments evaluate lexicon integration via the two distinct $\text{NorBert}_{3,\text{large}}$-based lemmatisation-rule and morphological classifier models, this implementation also handles sentence boundary and written standard (Nynorsk or Bokmål) detection.

Using a large model (323M parameters) for all of the mentioned tasks could become computationally intensive when processing large files. Therefore, during the implementation of the tagger, we have used a single base model and fine-tuned it with four classification heads for the

[7] https://huggingface.co/Humit-Oslo/humit-tagger-xs/tree/main

Input size	xs (cpu)	xs	small	base	large
1MB	00:56	00:55	00:59	01:10	02:00
10MB	06:13	04:32	04:50	06:18	13:57
50MB	30:01	19:16	20:50	27:55	58:34
100MB	58:35	39:13	41:35	57:42	2:11:51

Table 7: Execution durations of different sizes of the trained models on various sizes of input. Format: h:mm:ss

four different tasks mentioned here, including morphological and lemma-rule classification. Following our findings in this paper, we have implemented the lexicon in the tagger using the `model_lexicon_disamb` configuration. We have used various-sized NorBERT models (number of parameters from 15M to 323 M) as the base, enabling the tagger to run on a range of devices, including those with limited resources, such as in CPU-only mode with low RAM.

Table 7 gives an overview of the time required to run the tagger with varying input sizes and base models. When input is provided to the tagger, all classification heads are computed, although we use the tagger in a setting where only the lemmatisation and morphological tagging outputs are used. We tested the smallest model on a 14-core Intel Core i5 Linux-operating-system machine with 64 GB RAM, and we tested all models on an NVIDIA GeForce RTX 2080 GPU with 12 GB RAM. While our implementation tags a 1MB plain text file ranging from 56 seconds to 2 minutes, depending on the setting, Stanza (Qi et al., 2020), a Python NLP package based on neural networks, POS-tags and lemmatises the same text in 6 minutes on the same CPU and 3 minutes on the same GPU environment. This makes our implementation competitive in terms of speed with one of the widely-utilized lemmatisation and POS-tagger implementations. For usability, we have implemented the tagger as downloadable models in our HuggingFace repository[8], which requires only a few lines of code to run.

7. Conclusion and Future Work

Despite accuracy scores that are already in the high nineties, there is still scope for improving neural lemmatisation systems by combining them with high-quality lexical resources. Lemmatisation is an inherently difficult task for LLM-based systems because the output set, the vocabulary of the target language, is open-ended. The UD lemmatisation system of Straka et al. (2019) partly overcomes this by classifying into rewrite rules from form to lemma, rather than directly to lemmas, but the number of classes remain large, and performance is improved when handcrafted lexicons are used instead of the model whenever possible.

As we have seen above, this strategy noticeably reduces the number of true errors in the Norwegian data. However, when accuracy rates are as high as above 99%, it becomes very important to do careful error analysis. For example, around 10% of the errors are due to mistakes in the gold data. And many more errors are due to cases where multiple analyses would be acceptable.

This points to a general difficulty in integrating neural, data-driven systems with hand-crafted lexicons, namely that the training data for the neural system may rely on different annotation conventions from those of the lexical resources. As *Norsk ordbank* is a resource that is continually maintained by the Language Council of Norway, it is desirable to make it easy to integrate new words from this source. By contrast, the Norwegian Dependency Treebank is no longer actively maintained, which means that it will be a one-time effort to change its annotation practices to match those of *Norsk ordbank* (as long as this resource does not change its annotation practices).

Another line of work we will consider in the future is to increase the number of cases where the lexicon can be used by implementing compound analysis. As we have mentioned, compounding is a productive process in Norwegian and in many, probably most, cases, both compound members are in the lexicon.

Finally, with performance at such high levels, it will clearly be useful to construct new and more challenging datasets. This would include historical data, non-standard data with more frequent spelling mistakes, specialized language with rare terminology, etc.

8. Limitations

This work only deals with Norwegian. As such, the techniques and the configurations we use are only applicable for a high-resource language that has both enough text data to train a BERT model (Vaswani et al., 2017), enough annotated data to fine-tune this model for lemmatisation and morphological classification, *and* rich lexical resources such as a morphological lexicon. Very few languages have all of this. However, there is a lesson for planning the development of new resources for low-resource languages: rich, structured lexical information can still make an impact. As pointed out by one of the reviewers, there are languages with relatively few resources that still have extensive dictionaries. Many dictionaries will

[8] https://huggingface.co/Humit-Oslo/humit-tagger-xs/tree/main

not give all possible realisations of lemmas, as *Norsk ordbank* does, but even a simple dictionary can be used to filter the output of a machine learning system and thereby avoid e.g. phonotactically impossible lemmas that risk undermining human confidence in the output.

9. Bibliographical References

Tove Fjeldvig and Anne Golden. 1984. Automatisk rotlemmatisering (automatic root lemmatization) [in Norwegian]. In *Proceedings of the 4th Nordic Conference of Computational Linguistics (NODALIDA 1983)*, pages 84–103, Uppsala, Sweden. Centrum för datorlingvistik, Uppsala University, Sweden.

Emiliano Raul Guevara. 2010. NoWaC: a large web-based corpus for Norwegian. In *Proceedings of the NAACL HLT 2010 Sixth Web as Corpus Workshop*, pages 1–7, NAACL-HLT, Los Angeles. Association for Computational Linguistics.

Dag Haug, Ahmet Yildirim, Kristin Hagen, and Anders Nøklestad. 2023. Rules and neural nets for morphological tagging of Norwegian - results and challenges. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 425–435, Tórshavn, Faroe Islands. University of Tartu Library.

Janne Bondi Johannessen and Kristin Hagen. 2003. Parsing nordic languages (panola) norsk versjon. In Henrik Holmboe, editor, *Nordisk Sprogteknologi 2002*, pages 89–95. Museum Tusculanums Press.

Janne Bondi Johannessen, Kristin Hagen, André Lynum, and Anders Nøklestad. 2012. Obt+stat: A combined rule-based and statistical tagger. In Gisle Andersen, editor, *Exploring Newspaper Language*, pages 51–66. John Benjamins Publishing Company.

Thomas Karterud, Kristin Rogge Pran, and Mathilde Horvei. 2024. Språkrådets brukerog befolkningsundersøkelse. https://sprakradet.no/wp-content/uploads/Sprakradets-Bruker-og-befolkningsundersokelse-2024.pdf. Accessed: 2026-02-16.

Per E Kummervold, Javier De la Rosa, Freddy Wetjen, and Svein Arne Brygfjeld. 2021. Operationalizing a national digital library: The case for a Norwegian transformer model. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 20–29, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.

Arya D. McCarthy, Ekaterina Vylomova, Shijie Wu, Chaitanya Malaviya, Lawrence Wolf-Sonkin, Garrett Nicolai, Christo Kirov, Miikka Silfverberg, Sabrina J. Mielke, Jeffrey Heinz, Ryan Cotterell, and Mans Hulden. 2019. The SIGMORPHON 2019 shared task: Morphological analysis in context and cross-lingual transfer for inflection. In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 229–244, Florence, Italy. Association for Computational Linguistics.

Kirill Milintsevich and Kairit Sirts. 2021. Enhancing sequence-to-sequence neural lemmatization with external resources. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3112–3122, Online. Association for Computational Linguistics.

Lilja Øvrelid and Petter Hohle. 2016. Universal Dependencies for Norwegian. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1579–1585, Portorož, Slovenia. European Language Resources Association (ELRA).

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online. Association for Computational Linguistics.

David Samuel, Andrey Kutuzov, Samia Touileb, Erik Velldal, Lilja Øvrelid, Egil Rønningstad, Elina Sigdel, and Anna Palatkina. 2023. NorBench – a benchmark for Norwegian language models. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 618–633, Tórshavn, Faroe Islands. University of Tartu Library.

Per Erik Solberg, Arne Skjærholt, Lilja Øvrelid, Kristin Hagen, and Janne Bondi Johannessen. 2014. The Norwegian dependency treebank. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 789–795, Reykjavik, Iceland. European Language Resources Association (ELRA).

Milan Straka, Jana Straková, and Jan Hajic. 2019. UDPipe at SIGMORPHON 2019: Contextualized embeddings, regularization with morpho-

logical categories, corpora merging. In *Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 95–103, Florence, Italy. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

10. Language Resource References

National Library of Norway. 2014. *The Norwegian dependency treebank*. National Library of Norway. PID https://www.nb.no/sprakbanken/ressurskatalog/oai-nb-no-sbr-10/.

National Library of Norway. 2022a. *Norsk ordbank bokmål*. National Library of Norway. PID https://www.nb.no/sprakbanken/ressurskatalog/oai-nb-no-sbr-5/.

National Library of Norway. 2022b. *Norsk ordbank nynorsk*. National Library of Norway. PID https://www.nb.no/sprakbanken/ressurskatalog/oai-nb-no-sbr-41/.

National Library of Norway. 2023. *Norwegian UD treebank*. National Library of Norway. PID https://www.nb.no/sprakbanken/ressurskatalog/oai-nb-no-sbr-83/.

Quantification in Abstract Meaning Representation

Kiyong Lee and Chongwon Park
Korea University, Seoul, University of Minnesota Duluth
ikiyong@gmail.com, cpark2@d.umn.edu

Abstract

Quantification is common in text but underrepresented in AMR. It also strains the common conjunctive interpretation of AMR graphs, since universal quantification introduces scope-taking structure and variable binding that cannot be captured as a flat list of conjuncts. We propose an enriched AMR that supports quantificational meaning while preserving AMR's graph backbone. At the predicate level, we add QuantML-style features such as domain restriction, determinacy, distributivity, and involvement. At the discourse level, we add contextual constraints that encode scope and other discourse-sensitive conditions. The two levels follow the UMR architecture and are linked by shared identifiers. We map the enriched graphs to two-block logical forms: a minimal predicate structure of events and participants, plus a constraint block that relates them.

Keywords: QuantML, quantification, minimal logical form, predicate structure, discourse-level contextual constraint

1. Motivation, Aim, and Tasks

Since Montague (1974) and Bach et al. (1995), quantification has been a central and contested topic in natural language semantics. Quantificational expressions are pervasive in ordinary language and raise some of the most technically demanding problems in semantic analysis, including scope ambiguity, distributivity, and domain restriction. By contrast, AMR work has largely set aside quantification, a gap that can be detrimental, especially for downstream applications that require precise meaning representations. A further motivation is that, under a common baseline interpretation where an AMR corresponds to a conjunction of atomic conditions, universal quantification cannot be handled as just another conjunct, because it requires a conditional (implication) structure and explicit variable binding, giving rise to well-known scope and bound-variable problems (Bos, 2016, 2020).

This paper investigates how quantificational phenomena, including scope ambiguity and discourse-sensitive constraints on the interpretation of quantified expressions, can be represented by enriching Abstract Meaning Representation (AMR) with a quantification layer compatible with Uniform Meaning Representation (UMR). We adopt the ISO 24617-12:2025 QuantML feature inventory, including domain restriction, determinacy, distributivity, and involvement, and encode these features as annotations that can be mapped into enriched AMR structures.

For this purpose, this paper undertakes three specific tasks. The first task is to adopt the two-layer architecture of Uniform Meaning Representation (UMR, 2022), pairing a sentence-level AMR-style predicate-argument graph with a document-level component that encodes cross-sentential relations such as temporal and modal dependencies and coreference. The document-level component is linked to sentence-level nodes through shared identifiers (for example, by referencing sentence-indexed concept ID's), yielding an integrated representation that can support discourse-sensitive constraints on quantifier interpretation.

The second task is to define a set of discourse-sensitive constraints in the document-level component. The set includes constraints inferred from context and those triggered by quantificational features annotated at the sentence-level AMR nodes. The constraint formalism is inspired by Bos (2020)'s AMR$^+$ proposal, which preserves AMR's predicate argument backbone. We do not, however, follow Bos's way of marking those constraints by adding a logical layer with explicit indices.

The third task is to construct two-block logical forms, roughly corresponding to the two-level serialization of each annotation structure. The first block represents a (minimal) predicate structure, consisting of an event and a list of objects that participate in it, while the second block represents how all of the logical forms in the first block are related by contextual discourse constraints.

2. Main Issues

Two clarifications are needed before turning to the technical issues in this section. The first concerns why we keep the primary meaning representation graph-based and comparatively simple. The second concerns why we add an explicit logical interpretation layer rather than reading quantificational scope directly from the graph.

First, we treat enriched AMR as the primary interface because the paper is aimed at a represen-

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 104–113

tation suitable for corpus annotation and computational pipelines, not only for hand-built semantic analyses. AMR's practical value has been its compact predicate-argument graph backbone and its compatibility with existing annotation practice and parsers (Banarescu et al., 2019; Knight et al., 2020). If quantificational information is introduced in a way that forces pervasive reification, explicit variable bookkeeping, or global scopal decisions at every step, the representation becomes harder to annotate consistently and harder to learn from data. For this reason, we aim for a conservative extension strategy. That is, quantificational content is added as a small inventory of local features and discourse-level constraints that remain compatible with the two-layer UMR architecture (UMR, 2022) and that can be mapped from the QuantML annotation scheme and inventory (Bunt et al., 2018, 2022; ISO, 2025). This design keeps the graph readable, keeps the enrichment optional for applications that do not need it, and preserves interoperability with existing AMR tooling.

Second, the logical forms in this paper are neither a competing representation layer nor an additional annotation target. They are used as a reference semantics for the enriched graphs. Once the representation is enriched with meaning-bearing features and discourse constraints, readers need an explicit statement of what those annotations commit us to in interpretation. The logical forms provide that statement in a standard, checkable metalanguage. They make scope, domain restriction, and distributive conditions explicit enough to (i) verify that the enriched graphs distinguish the intended readings, (ii) validate conversions between QuantML-style annotation structures and AMR-style graphs, and (iii) support accuracy claims about the representation by comparing whether two analyses yield the same or different truth-conditional commitments (Bos, 2016, 2020).

A key reason for spelling out the reference semantics is that a common baseline interpretation of AMR treats an AMR graph as a flat conjunction of atomic conditions (a list of conjuncts). This works well for many positive, existential-like cases, but it breaks down once scopal operators are needed. In particular, universal quantification requires a conditional (implication) structure, which introduces explicit variable binding and resolves the bound-variable and scope problems that arise under a purely conjunctive view (Bos, 2016, 2020). An equivalent option is to compile universals into negation and existential quantification (using $\forall x\, \phi \equiv \neg\exists x\, \neg\phi$), but this still requires scope-taking operators and therefore cannot be recovered from conjunction alone (Bos, 2016, Section 5). The logical forms in this paper make these scope and binding commitments explicit while keeping the annotation-layer graphs lightweight.

2.1. Simplicity vs. Adequacy

The first design issue concerns the balance between notational simplicity and representational adequacy in meaning representation. This section compares two concrete syntaxes for encoding quantificational analyses: the XML-based representation format, used in ISO QuantML (ISO, 2025), and the PENMAN notation, used to serialize AMR-style rooted, directed graphs.

Within the ISO 24617 Semantic Annotation Framework, QuantML specifies an XML-based representation format for annotation structures. This design choice follows the broader ISO practice of using XML as a uniform concrete syntax across annotation standards. It is largely for syntactic interoperability and reuse of existing XML tooling, rather than for compactness or ease of direct human authoring. At the same time, QuantML distinguishes an abstract syntax from its XML concrete syntax, which in principle allows alternative serializations that preserve the same underlying annotation structures (Bunt, 2010; ISO, 2016).

For dense semantic annotation and integration with AMR-style graphs, however, XML can be prohibitively verbose. Even for small fragments, an XML encoding like Example (1) requires nested elements, semantic categories encoded as tag names, and multiple identifiers for cross-references. These devices make entity structures and link structures explicit, but they also obscure the underlying graph and complicate both manual inspection and conversion into a graph-based meaning representation.

Consider Example (1) that annotates a fragment of text in XML:

(1) Sentence: Dogs bark.
Segmented: Dogs$_{w1}$ bark$_{w2}$.

```
<ENTITY xml:id="x1"
target="#w1" pred="dog"/>
<EVENT xml:id="e1" target="#w2"
pred="bark"/>
<SRLINK eventID="#e1"
participantID="#x1"
relType="agent"/>
```

In contrast, AMR encodes the same semantic content for the purposes of sentence interpretation, but does so in a more compact and direct notation.

(2) :: Sentence: Dogs bark.
:: Interpretation: For the present comparison, the example introduces a barking event with a dog participant.
:: id: amr001KL,
:: Data-creation time: 2026-01-23

```
(b / bark-01
   :agent (d / dog))
```

Although the surface format differs substantially, the AMR representation in Example (2) introduces the same core semantic objects and relations: an eventuality of barking and a dog participant linked to that event. As a result, both representations support the same core predicative interpretation of the sentence, despite their different levels of structural and notational complexity.

By comparing the two representations, (1) and (2), we observe that several components of the XML representation can be removed to reduce the size of the AMR representation. First, all of the tag names, such as `ENTITY`, `EVENT`, and `SRLINK`, need not be represented in AMR. Second, the link for participant-role assignment is unnecessary; it is built into the AMR structure.

2.2. Logical Forms and their Interpretation Model

The second issue concerns how to interpret enriched AMR structures for quantification.

(3) :: Sentence: All dogs bark.
:: Interpretation:
There are dogs, and all of them bark.
:: id: amr002KL,
:: Data-creation time: 2026-03-22

```
(b / bark-01
   :agent (d / dog
             :num PL
             :quant A))
```

Here, PL and A are logical constants standing for plurality and universal quantification. A minimal predicate structure is obtained by abstracting away from these two feature specifications.

2.2.1. Minimal Logical Form Representing a Predicate Structure

To this end, we first derive logical forms directly from minimal AMR predicate structures, such as (4).

(4) Minimal Logical Form
$\{b, d\}$
$[instance(b, bark),$
$instance(d, dog),$
$agent(b, d)]$

In AMR, b and d are treated as variables, but they can be understood as discourse referents in DRS (Kamp and Reyle, 1993). The comman (,) stands for logical conjunction ($\wedge$). The relation $instance$ or i, used in some places instead, is a convenient metalanguage predicate that links a discourse referent to an event predicate such as *bark* or an entity predicate such as *dog*.[1]

This is a *minimal logical form*, representing a predicate structure with its participants. It is minimal in two senses: first, it abstracts away from the feature specifications involving PL for plurality and A for universal quantification; second, it supports the construction of a minimal interpretation model in the spirit of Bos (2003) and Blackbun and Bos (2003); Blackburn and Bos (2004), based on the predicate structure contributed by AMR.

2.2.2. Minimal Model for Interpretation

In set-theoretic terms, a minimal interpretation model based on the AMR predicate structure (3), abstracted away from `:num PL` and `:quant A`, can be constructed as a tuple $<D, v>$, such that

(5) D=$\{b, d\}$
$v(instance)$=$\{(b, bark), (d, dog)\}$
$v(agent)$=$\{(b, d)\}$

The discourse domain D has two discourse referents, b and d. The semantic values of the two relations $instance$ and $agent$ are each specified as a set of pairs. This is a minimal model that satisfies each formula in (4), and thus the entire conjunctive form. It supports only the described situation: a barking event and a dog participant, with no further commitments beyond what the graph introduces.

Intuitively, the resulting minimal model describes a very small world, which we call the *described situation* (Barwise, 1981). The model is minimal in the sense that it does not commit us to any additional irrelevant individuals or facts beyond those required by the AMR structure itself. This paper adopts such a model construction for semantic interpretation.

2.2.3. Extension

The minimal model presented in (5) may appear trivial, but it is not fixed in a closed world. It may be extended when later discourse, contextual updates, or additional constraints require new individuals or additional relations. For example, suppose a situation is described in which two people are each drinking a beer. If subsequent discourse introduces the possibility that more people arrive and more beer is ordered, the minimal model can be extended by adding more people and more beer. In this way, the model functions as a small, incrementally extensible reference structure for interpretation.

Consider a simple example:

[1] Here $instance$ or i is used only in the reference semantics. It links discourse referents to the event or entity concepts introduced in the graph.

(6) :: Sentence: Three dogs are barking.
:: Interpretation: There are three dogs, and they are barking.

```
(b / bark-01
   :agent (d / dog
             :num 3)
   :aspect prog)
```

We construct a minimal model form by abstracting away from `:aspect prog` while retaining the cardinality information contributed by `:num 3`.

(7) Logical Form $\{b, x\}$
$[instance(b, bark)$,
$instance(x, dog)$,
$agent(b, x)$,
$num(x)$=3]

Here, $num(x)$=3 is a short-hand representation, as defined:

(8) For any variable x, natural number n, and non-emptyset X,
$num(x) \geq n =_{def} x \in X$ such that $|X| \geq n$.

To interpret Logical Form (7), a minimal model is constructed as a tuple $<D, v>$, where D is the discourse domain and v assigns a semantic value to each of the binary relations, $instance$ and $agent$, such that we have:

(9) Extended Minimal Model
$D = \{b\} \cup \{x | x \in X, |X|=3\}$,
$v(instance)$=$\{(b, bark), (x, dog)\}$,
$v(agent)$=$\{(b, x)\}$

The variable x ranges over the set X, the cardinality of which is 3. This model thus supports a situation in which there are three agents, which are dogs, that bark.

2.2.4. Two-Block Representation of Logical Forms for Contextual Constraints

The proposed framework, in which AMR is enriched with UMR-style contextual constraints, yields a two-block representation for logical forms in cases (at least) involving quantification. Each AMR-based predicate structure contributes a minimal logical form in Block 1. In contrast, the discourse-level constraints are stated as quantified logical forms in Block 2. To interpret the resulting representation, we treat Block 1 (minimal structure) as a reference basis: formulas in Block 2 may contain variables that are not explicitly bound there, but that are intended to refer back to discourse entities introduced in Block 1. Operationally, such variables are resolved via Block 1 and bound at the appropriate scope in Block 2. In this respect, the overall structure is close to DRS (Kamp and Reyle, 1993), which combines a set of atomic conditions with a quantificational component.

(10) Restricted Discourse Domain:
A minimal model M_q for quantification is a tuple $<D, r, v>$, where
D is a non-empty domain of discourse, r is a restriction on D such that the restricted domain $D_r \subseteq D$, and v is a semantic valuation.

Restricted quantificational domains are introduced by the relevant noun phrase or, more often, contextually. The valuation v is a denotation function that satisfies the concept instances and relations contributed by the predicate-focused AMR structure.

(11) Logical Form:
Block 1 (AMR-based minimal structure):
$\{e, x\}$
$[instance(e, protest)$,
$instance(x, student)$,
$num(x) \geq 2$,
$dom(x) \cap r(x)$,
$agent(e, x)]$
Block 2 (contextually constrained quantificational structure):
$\forall y$[y=x→
$\exists e_1[e_1 \sqsubseteq e \wedge agent(e_1, y)]]$

The representation in (11) lists the discourse entities $\{e, x\}$ that populate the minimal model, together with the conditions contributed by the AMR structure. The minimal model itself does not bind these variables; it records them as available referents. Block 2 then states the quantificational constraint over those referents.

The identity y=x in Block 2 is a *reference identifier*. It links Block 2 to the occurrences of x in Block 1, Predicate-focused minimal logical form. As defined in (8), the variable x is a variable in a set X the cardinality of which is 3. Hence, the quantificational form in Block 2 in (11) be rephrased as in (12) below:

(12) $\exists e[i(e, protest)$,
$\forall x[[i(x, student), x \in X, |X| \geq 2] \rightarrow$
$\exists e_1[e_1 \sqsubseteq e, agent(e_1, x)]]]$,
where the relation i again stands for the relation $instance$.

The effect of resolving the references to Block 1 can be made explicit by rewriting Block 2 so that the relevant discourse referents are existentially introduced along with the conditions that characterize them in the minimal model. This yields a compiled form such as (13):

(13) Block 2: Quantificational Structure:
$\forall y[y$=x
$\exists\{e_1, e\}[e_1 \sqsubseteq e \wedge i(e, protest) \wedge agent(e_1, y)]]$

Either representation corresponds to the same interpretive idea. One may interpret the original

two-block structure by allowing Block 2 to refer back to Block 1, or one may compile that reference step into a single quantified formula as in (13). The two-block format is useful because it separates (i) the AMR-contributed described situation from (ii) the additional quantificational constraints imposed by the discourse level.

The distributive effect can be made more explicit by quantifying over event parts and requiring each member of the student plurality to participate in an appropriate subevent:

(14) Logical Forms:
a. Block 1: Preamble
$\{e, x\}$
$[i(e, protest)$,
$i(x, student)$,
$num(x) \geq 2$,
$dom(x) \cap r(x)$,
$agent(e, x)]$
b. Block 2, referring to Block 1:
$\forall y[y{=}x \rightarrow \forall e_1[e_1 \sqsubseteq e \rightarrow agent(e_1, y)]]$

The logical form $dom(x) \cap r(x)$ in Block 1 represents the restricted domain of x for the universal quantification in Block 2. The quantified logical form in (b) captures the distributive effect by quantifying over event parts e_1 such that $e_1 \sqsubseteq e$ and by requiring $agent(e_1, y)$ for each such part. As before, the variables x and e are supplied by the discourse entities introduced in the first block.

In Section 3, we further discuss how this conversion can be triggered.

3. Constructing the Architecture of AMR Linked to UMR

Uniform Meaning Representation (UMR (2022), 0.9 Specification) extends AMR by pairing a sentence-level graph with a document-level graph that supports cross-sentential phenomena. UMR retains AMR's predicate-argument structures at the sentence level, but augments them with document-level annotations that link events and entities across sentences and encode temporal, coreferential, and modal information. Lee et al. (2025) suggest calling this UMR *document level* a *discourse level*, since many of these relations are defined over broader context rather than within a single sentence and can interact with phenomena such as scope and temporal interpretation.

Building on Pustejovsky et al. (2019), which addressed quantification and scope in AMR, we propose to join two levels: a predicate-structure level (P-level) and a discourse level (D-level). These two levels are connected by a root node, here called *Meaning*, as shown in Figure 1. The resulting representation is a single-rooted, labeled, directed graph with reentrancies; like standard AMR, it need not be a tree, as shown in Figure 2.

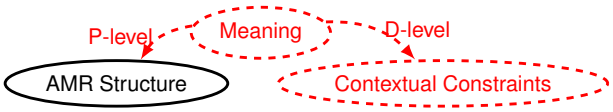


Figure 1: Architecture of the Enriched AMR

The syntax of both the predicate and discourse levels is stated in a PENMAN-style format compatible with current AMR conventions (Knight et al., 2020). We thus retain the same type of syntax for both levels. In the graph, each concept c in C, including the root, corresponds to a node. Relations label directed edges between nodes, where each edge goes from a parent node to a child node. A single parent node may have multiple children. The overall structure is interpreted as a labeled, directed graph whose connectivity is determined by these parent–child relations.

In the example below, each parenthesized variable introduces a node, and each colon-prefixed label (e.g., `:agent`, `:scope`) introduces a directed relation from the current node to its child, whether at the P-level or the D-level.

(15) :: Sentence: All the students protested.
:: Interpretation: There were students, and all of them participated in a protest.
:: id: 001, 2026-02-05

```
(r / meaning
   :P-level
     (e / protest-01
       :agent (x / student
                 :num PL
                 :dom (d / restr)
                 :quant A))
   :D-level
      (c / constraints
         :domain (ex / entities
                     :op1 e
                     :op2 x)
         :scope (co / conditional
                     :op1 x
                     :op2 e)))
```

At the discourse level, constraints for interpretation specify the domain of discourse and the scopal relation of universal quantification. First, the domain lists the event variable e and the plural participant variable x. Second, the scope relation states that the student plurality takes scope with respect to the protest event. The corresponding two-block logical form can be stated as follows:

(16) a. $\{e, x\}$
$[instance(e, protest)$,

$instance(x, student)$,
$num(x) \geq 2$,
$dom(x) \cap r(x)]$
b. $\forall y[y{=}x \rightarrow$
$\exists e_1[e_1 \sqsubseteq e \wedge agent(e_1, y)]]$

Here, $num(x) \geq 2$ is shorthand for plurality: x denotes a plural discourse domain or for the range of x with at least two members.

The detailed annotation at the predicate and discourse levels can be incorporated into the general graph structure of the enriched AMR linked to the discourse level of UMR, following Pustejovsky et al. (2019) that combined two levels to treat scope ambiguity.

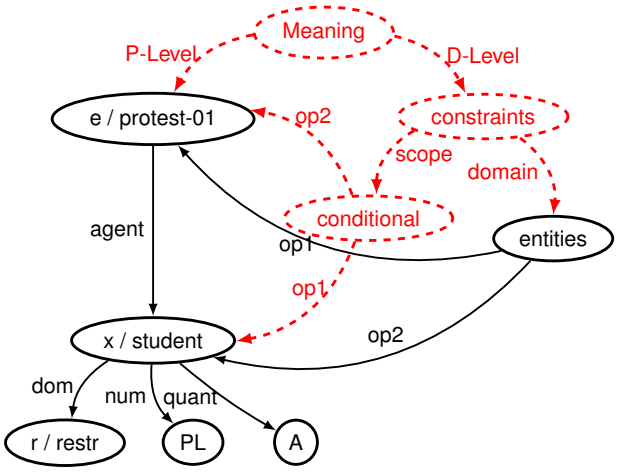


Figure 2: Enriched AMR with Quantification

The encoding of the contextual constraints, represented on the graph, may be treated by the indexing mechanism referred to in Bos (2020), but we have taken a different option.

4. Validation with Further Illustrations

4.1. Overview

This section informally validates the proposed two-block reference semantics with further illustrations. Various discourse constraints, such as scope involving quantification or presuppositional existence associated with definite descriptions or proper names, can be represented by incorporating Bos (2020)'s contextual constraints into the *discourse level*. These contextual constraints allow AMR to avoid encoding some information redundantly. With conditional indexing for universal quantification, Bos even argues that the traditional `:quant all` marking can be dispensed with. In the present proposal, however, it is better to retain overtly expressed information, such as determinacy, rather than relying on contextual information alone.

4.2. Definite Descriptions: Negation and Presupposition

Definite descriptions, such as *my cat* and *the rats*, are known to trigger existential presupposition. Consider how definite description phrases interact with negation in a sentence.

(17) :: Sentence: My cat didn't chase out all the rats from the house.
:: Interpretation: I had a cat, and there were rats in my house, but it is not the case that the cat chased all of the rats out of the house.
:: DCT 2026-02-11

Sentence (17) contains three definite descriptions, *cat*, *rats*, and *house*. Encoding each of them with `:determinacy (d / def)`, as in QuantML, presupposes their existence.

(18)
```
(e / chase-01
   :polarity -
   :agent (c / cat
             :poss (m / me)
             :determinacy
               (d1 / def))
   :theme (r / rat
             :determinacy
               (d2 / def))
   :source (h / house
             :determinacy
               (d3 / def))
   :goal (o / out))
(cn / constraints
   :presuppose (ex / exists
                  :agent c
                  :theme r
                  :source h)
   :scope (w1 / wide
             :op1 -
             :op2 e)
   :scope (w2 / wide
             :op1 r
             :op2 e))
```

Annotation (18) yields Logical Form (19):

(19) Logical Form:
a. Block 1:
$\{e, c, m, r, h, o, d_1, d_2, d_3\}$
$[instance(e, chase_{01})$,
$instance(c, cat), instance(d_1, def)$,
$instance(m, me)$,
$instance(r, rat), instance(d_2, def)$,
$instance(h, house), instance(d_3, def)$,
$instance(o, out)$,
$polarity(e, -)$,
$arg_0(e, c), arg_1(e, r)$,
$source(e, h), goal(e, o)]$
b. Block 2, referring to Block 1:

$\neg\forall y[y{=}r \rightarrow$
$[instance(e, chase_{01}), agent(e, c),$
$theme(e, y), source(e, h), goal(e, o)]]$

The first part of Logical Form (19) states that the described situation introduces an event of chasing and the discourse entities corresponding to the cat, the rats, and the house. The second part states the weak reading of the negated universal: it is not the case that every relevant rat was chased out of the house by the cat. If the negation operator were placed elsewhere, the truth conditions would change while the existential presuppositions remain in force.

4.3. Subevents

The AMR Guidelines (Banarescu et al., 2019) list `:subevent` and `:part` among the non-core relations. These relations can help encode the internal structure of complex events such as sharing something edible or a house, the Olympic Games with many component events, or wars with dispersed battles.

Consider a case of two women sharing a piece of pizza, taken from Bunt and Lee (2025). The AMR annotation is simplified by using the relation `:subevent`.

(20) :: Sentence: Two women shared a pizza.
:: Interpretation: There were two women and a pizza, and they shared it, each eating part of it.
:: CDT-2026-01-28:T20:10

```
(s / share-01
   :agent (w / woman
             :quant 2)
   :theme (p / pizza
             :quant 1)
   :manner collective
   :subevent
     (e / eat-01
       :agent w
       :theme (p1 / pizza
                 :part-of p)
       :manner individual))
(c / constraints
   :scope (w1 / wide
             :op1 p
             :op2 w)
   :scope (w2 / wide
             :op1 p1
             :op2 e))
```

In the intended scenario, sharing the pizza is modeled as including eating subevents. The eating event is therefore annotated as a subevent of the sharing. Logical Form (21) below is derived from Annotation (20).

(21) Logical Form:
a. Block 1: Preamble
$\{s, w, p, e, p1\}$
$[instance(s, share_{01}),$
$instance(w, woman), quant(w, 2),$
$instance(p, pizza), quant(p, 1),$
$agent(s, w), theme(s, p),$
$manner(s, collective),$
$instance(e, eat_{01}),$
$subevent(e, s), agent(e, w),$
$instance(p1, pizza), part\_of(p1, p),$
$theme(e, p1), manner(e, individual)]$
b. Block 2, referring to Block 1:
$\forall e_1[e_1 = e \rightarrow [instance(e_1, eat_{01}),$
$agent(e_1, w), theme(e_1, p1)]]$

Logical Form (b) again refers to the preamble for the variables supplied by Block 1.

4.4. Distributivity and Coreference

Here is another example from QuantML (ISO (2025), Example 5) involving distributivity and coreference.

(22) :: Sentence: The men had a beer, and they carried the piano upstairs.
:: Interpretation: There were some men and a piano. Under the intended reading, each man drank a beer, and the same group then carried the piano upstairs together.
:: id: k006, DCT 2025-01-04

```
:predicate-level
 (d / drink-01
    :agent (m / man
              :num PL)
    :theme (b / beer
              :quant 1)
    :manner individual)
 (c / carry-01
    :agent (m1 / man
              :num PL)
    :theme (p / piano
              :quant 1)
    :goal (u / upstairs)
    :manner collective)
:discourse-level
 (cn / constraints
     :presupposition
         existential
     :temporal (bf / before
                  :op1 d
                  :op2 c)
     :scope (sc / conditional
                :op1 m
                :op2 d)
     :coreference
         (id / identical
            :op1 m :op2 m1))
```

The individual distributivity of the drinking event triggers the scope constraint over the men, while the carrying event is marked as collective. The coreference relation states that the men who drank are the same men who later carried the piano.

(23) Logical Forms:
a. $\{d, m, b, c, m1, p, u\}$
$[instance(d, drink_{01}),$
$instance(m, man), num(m) \geq 2,$
$instance(b, beer), quant(b, 1),$
$instance(c, carry_{01}),$
$instance(m1, man), num(m1) \geq 2,$
$instance(p, piano), quant(p, 1),$
$instance(u, upstairs),$
$agent(d, m), theme(d, b),$
$manner(d, individual),$
$agent(c, m1), theme(c, p),$
$goal(c, u),$
$manner(c, collective)]$
b.
$\forall y[y{=}m \rightarrow$
$\exists e_1[e_1 \sqsubseteq d, agent(e_1, y),$
$\tau(d) \prec \tau(c), m{=}m_1)]$,
where τ maps an event to the time of its occurrence (event time).

This logical form makes the distributive reading explicit for the drinking event while keeping the collective carrying event and the coreference relation at the discourse level.

5. Conclusion

Quantificational meaning is pervasive in text, but it remains difficult to express in standard AMR under the common "flat conjunction" interpretation. In particular, universal quantification and distributive readings require scope-taking structure and variable binding that cannot be recovered from a mere list of conjuncts. This paper presented a conservative extension of AMR that makes quantificational commitments explicit while preserving AMR's practical graph backbone and remaining compatible with the two-layer architecture of UMR..

The paper makes five concrete contributions. First, we proposed an AMR-compatible representation in which a sentence-level predicate-argument graph (P-level) is paired with a document/discourse component (D-level) that records contextual constraints relevant to quantifier interpretation. The two levels are explicitly *linked* by shared identifiers, allowing discourse constraints (e.g., scope) to target the same event and participant nodes introduced at the predicate level.

Second, we showed how key ISO 24617-12:2025 QuantML dimensions (including *domain restriction*, *determinacy*, *distributivity*, and *involvement*) can be represented as local, meaning-bearing annotations on AMR nodes/edges in PENMAN, avoiding the verbosity of the XML concrete syntax while preserving the underlying annotation content.

Third, building on the spirit of contextual constraints (e.g., AMR$^+$), we defined D-level constraint structures (e.g., conditional scope constraints, presuppositional existence constraints, and cross-sentential linking) that encode scopal dependencies and discourse-sensitive conditions without introducing a separate layer of explicit index bookkeeping inside the graph.

Fourth, we provided a systematic mapping from enriched graphs to a two-block logical form: (i) a *minimal model* block that records the described situation (events and discourse referents contributed by the AMR predicate structure), and (ii) a *constraint* block that states quantificational and discourse-sensitive requirements (e.g., universal/conditional structure, presuppositions, temporal constraints) while allowing controlled reference back to Block 1.

Finally, through detailed examples, we illustrated how the proposal captures (i) restricted universals as conditional scope constraints, (ii) interactions between definiteness, negation, and presuppositional existence, and (iii) distributive interpretations that require event-structural enrichment (e.g., event-part quantification and subevent structure) while maintaining a single graph-based interface.

Overall, the proposal aimed to make quantification representable and checkable in AMR-style annotation. QuantML feature content remains locally readable at the predicate level, and scope-taking and discourse-sensitive commitments are stated explicitly at the discourse level to support interpretation and downstream reasoning.

Next steps include (i) expanding and systematizing the discourse-level constraint inventory (including additional triggers from QuantML features), (ii) developing annotation guidelines and consistency criteria for the enriched representation, and (iii) conducting empirical evaluation, including annotation studies and experiments on parsing and downstream tasks that are sensitive to quantificational distinctions (e.g., inference under scope and distributivity).

6. Acknowledgements

We thank the three anonymous reviewers for their detailed and constructive comments, which have substantially improved the paper. We have addressed most of their suggestions in the present revision. We also owed a great deal to three scholars: Harry Bunt, Johan Bos, and James Pustejovsky, for their original ideas, which we tried to incorporate into this seminal work. A few broader issues,

mainly concerning semantic interpretation, are beyond the scope of this paper and are left for future work.

7. Bibliographical References

Emmon Bach, Eloise Jelinet, Angelica Kratzer, and Barbara H. Partee, editors. 1995. *Quantification in Natural Language*. Kluwer Academic Pubishers, Dordrecht.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2019. *Abstract Meaning Representation (AMR) 1.2.6 Specification*. The DMR Working Group.

Jon Barwise. 1981. Scenes and other situations. *The Journal of Philosophy*, 78(3):369–397.

Patrick Blackbun and Johan Bos. 2003. Computational semantics. *Theoria*, 18(1):27–45.

Patrick Blackburn and Johan Bos. 2004. *Represenging and Inference for Natural Language: A First Course in Computational Semantics*. Center for the Study of Language and Information, Stanford, CA.

Johan Bos. 2003. Exploring model building for natural language understanding. In *ICoS-4. Inference in Computational Semantics. Workshop Proceedings*, pages 41–55.

Johan Bos. 2016. Squib: Expressive power of abstract meaning representations. *Computational Linguistics*, 42(3):527–535.

Johan Bos. 2020. Separating argument structure from logical structure in AMR. In *Proceedings of the 2nd Interantional Conference on Designing Meaning Representations*, pages 13–20, Barcelona, Spain (online), December 13, 2020.

Harry Bunt. 2010. A methodology for designing semantic annotation languages exploiting syntactic-semantic iso-morphisms. In *Proceedings of ICGL 2010, the Second International Conference on Global Interoperability for Language Resources*, pages 29–45, City University of Hong Kong.

Harry Bunt, Maxime Amblard, Johan Bos, Karën Fort, Bruno Guillaume, Philippe de Groote, Chuyuan Li, Pierre Ludmann, Michel Musiol, Siyana Pavlova, Guy Perrier, and Sylvain Pogodalla. 2022. Quantification annotation in ISO 24617-12, second draft. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3407–3416, Marseille, France. European Language Resources Association.

Harry Bunt and Kiyong Lee. 2025. The representation of QuantML annotations in UMR – an exploration. In *Proceedings of the 21th Joint ACL–ISO Workshop on Interoperable Semantic Annotation (ISA-21)*, Heinrich Heine University Düsseldorf, Germany. IWCS 2025.

Harry Bunt, James Pustejovsky, and Kiyong Lee. 2018. Towards an ISO standard for the annotation of quantification. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*, pages 1787–17949, Miyazaki, Japan. ELRA.

ISO. 2016. *ISO 24617-6:2016, Language resource management – Semantic annotation framework (SemAF) – Part 6: Principles of semantic annotation (SemAF Principles)*. The International Organization for Standardization, Geneva. Project leader: Harry Bunt.

ISO. 2025. *ISO 24617-12:2025 Language resource management – Semantic annotation framework (SemAF) – Part 12: Quantification*. The International Organization for Standardization, Geneva. Project leader: Harry Bunt.

Hans Kamp and Uwe Reyle. 1993. *From Discourse to Logic: An Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and DRT*. Kluwer, Dordrecht.

Kevin Knight, Bianca Badarau, Laura Baranescu, Claire Bonial, Madalina Bardocz, Kira Griffitt, Ulf Hermjakob, Daniel Marcu, Martha Palmer, Tim O'Gorman, and Nathan Schneider. 2020. *Abstract Meaning Representation Release 3.0*. Linguistic Data Consortium, Philadelpia, PA. Techinical Report LDC2020T02.

Kiyong Lee, Harry Bunt, James Pustejovsky, Alex C. Fang, and Chongwon Park. 2025. Representing ISO-annotated dynamic information in UMR. In *Proceedings of the Sixth International Workshop on Designing Meaning Representations*, pages 49–58, Prague, Czechia. Association for Computational Lingustics.

Richard Montague. 1974. The proper treatment of quantification. In *Formal Philosophy: Selected Papers of Richard Montague*, New Haven and London. Yale University Press.

James Pustejovsky, Nianwen Xue, and Kenneth Lai. 2019. Modeling quantification and scope in abstract meaning representations. In *Proceedings of the First International Workshop on*

Designing Meaning Representation, pages 28–33. Association for Computational Linguistics. Florence, Italy, August 1, 2019.

Working Group UMR. 2022. *Uniform Meaning Representation (UMR) 0.9 Specification*. UMR Working Group for Guidelines.

Improving Slovene Language Models for Lexicographic Question Answering through Continued Pretraining and Instruction Fine-Tuning

Timotej Knez, Slavko Žitnik
University of Ljubljana, Faculty of Computer and Information Science
Večna pot 113, Ljubljana, Slovenia
{timotej.knez, slavko.zitnik}@fri.uni-lj.si

Abstract

This paper presents a two-stage training approach to improve the performance of Slovene large language models on lexicographic question-answering tasks. We developed a comprehensive lexical pretraining corpus containing 356,294 Slovene word entries. We constructed the corpus by converting structured data from multiple lexicographic sources into markdown format. Additionally, we created a question-answering dataset with 16,508 QA pairs from diverse sources, including automatically generated questions, a linguistic advisory portal, and community forums. Using the Slovenian GaMS model (based on Gemma 2 9B) and GaMS 3 model (based on Gemma 3 12B), we performed continued pretraining on the lexical corpus followed by instruction fine-tuning with our QA dataset combined with translated general-domain questions. We compared results to different model configurations. Our results demonstrate significant improvements (text similarity increasing from 0.226 to 0.542, BERTScore F1 of 0.915) in answering Slovene lexicographic questions, validating the effectiveness of domain-specific continued pretraining for low-resource languages.

Keywords: Slovene, language models, lexicography, continued pretraining, instruction fine-tuning, question answering

1. Introduction

Lexicographic knowledge plays a fundamental role in natural language processing, providing essential information about word meanings, morphological forms, usage patterns, and semantic relationships (Horák and Rambousek, 2017). While large language models (LLMs) have demonstrated remarkable capabilities across various tasks, they often struggle with specialized lexicographic questions, particularly for low-resource languages where pretraining data may be limited and less diverse (Merx et al., 2024). For Slovene, a South Slavic language with complex morphology and relatively scarce digital resources, this challenge is especially pronounced.

The gap between general pretraining corpora and domain-specific lexicographic knowledge presents a significant obstacle. Standard pretraining approaches expose models to broad linguistic patterns but may not adequately capture the structured, detailed information contained in lexicographic databases. This is particularly problematic for questions that require precise knowledge of word forms, definitions, collocations, synonyms, and morphological paradigms, information that is systematically organized in lexicographic resources but may be underrepresented or scattered across general text corpora.

This paper addresses a research question: Can structured lexicographic data improve Slovene LLM performance through a combination of continued pretraining and instruction fine-tuning? We hypothesize that converting structured lexicographic resources into natural language text and using them for continued pretraining will inject domain-specific knowledge into the model, while subsequent instruction fine-tuning will align the model with question-answering tasks.

In summary, our experiments show that lexicographic adaptation is highly effective: direct lexicographic instruction fine-tuning (Lex-FT) produced the best overall results, substantially improving Slovene lexicographic QA quality over baseline models, while continued pretraining provided limited additional gains and general benchmark performance remained broadly stable.

Our main contributions are:

- A lexical pretraining corpus containing 356,294 Slovene word entries synthesized from multiple lexicographic sources and structured in markdown format
- A comprehensive QA dataset with 16,508 question-answer pairs from diverse sources, including automatically generated questions, linguistic advisory portals, and community forums
- A two-stage training methodology combining continued pretraining with instruction fine-tuning, applied to the GaMS model (based on Gemma 2 9B architecture)
- Comprehensive evaluation demonstrating significant improvements in answering Slovene

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 114–123

lexicographic questions while maintaining general language capabilities

2. Related Work

2.1. Slovene Lexical Resources

Slovene benefits from a comprehensive ecosystem of digitized lexical resources developed primarily by the Center for Language Resources and Technologies (CJVT) at the University of Ljubljana. These resources collectively provide definitions, usage examples, word forms, and collocations that form the foundation of our lexicographic knowledge base.

The Digital Dictionary Database of Slovene (DDDS) serves as the central lexicographic repository. Linked to DDDS, the Open Slovene WordNet (OSWN) provides a semantic network of Slovene synsets: OSWN 1.0 contains approximately 95,262 synsets covering 164,904 word forms, derived from Princeton WordNet with Slovene translations (Čibej et al., 2023). CJVT also maintains the Thesaurus of Modern Slovene, the largest open Slovene synonyms collection, which is fully digital and crowdsourced (Krek et al., 2023). This thesaurus interlinks with other resources and enables users to compare headwords and synonyms along collocational profiles with corpus examples.

For contextual information, the Collocations Dictionary of Modern Slovene provides 78,046 headwords with 4.4 million collocation pairs and approximately 14.45 million usage examples (Kosem et al., 2023). For morphology, Sloleks 2.0 (Slovene Morphological Lexicon) covers 100,802 lemmas and about 2.79 million inflected word forms with automatic stress annotation (Dobrovoljc et al., 2019). Together, these resources provide nearly all facets of a dictionary entry: sense definitions, usage examples, inflectional paradigms, and collocational patterns, making them good knowledge base for language model enhancement.

2.2. Slovene Large Language Models

Recently, CJVT has developed Slovene-specific LLMs. The GaMS family ("Generative Model for Slovene") includes models continually pretrained on Slovene and related South Slavic data (Vreš et al., 2024). Vreš et al. (2024) created GaMS-1B by continuing pretraining on the English OPT model with Slovene data. Larger models followed: GaMS-9B (9B parameters) and GaMS-27B (27B parameters) are based on Google's Gemma 2 multilingual architecture, continually pretrained on Slovene plus Croatian, Serbian, Bosnian, and English (CJVT, 2024). More recently, CJVT released GaMS3-12B-Instruct, the next-generation GaMS based on Google's Gemma 3 architecture (CJVT, 2026). These models serve as powerful Slovene backbones for further fine-tuning. In this work, we use GaMS models as our base to specialize in lexicographic data.

2.3. LLMs for Lexicography and Lexical Semantics

There is growing interest in using LLMs for lexicographic tasks. Pham et al. (2025) found that ChatGPT's word definitions are highly accurate, comparable to traditional dictionaries. Similarly, Lew (2023) showed that ChatGPT could generate COBUILD-style dictionary entries for English, with AI-generated sense definitions rated on par with human lexicographers' definitions, though example sentences scored lower. These results demonstrate LLMs' raw capacity for lexicographic output but also their limitations, particularly in producing illustrative examples. Importantly, both studies suggest that fine-tuning could improve results.

On the other hand, several studies have shown that injecting structured lexical knowledge into LLMs via fine-tuning can boost performance on lexical-semantic tasks. Moskvoretskii et al. (2024) created TaxoLLaMA by compiling an instruction-tuning dataset from English WordNet hypernym relations, then fine-tuning LLaMA-2 on this data, achieving state-of-the-art results on taxonomy enrichment and hypernym discovery tasks. This illustrates that feeding WordNet-style definitions and relations into an LLM can make its implicit word knowledge more explicit and usable.

For Slovene specifically, Škvorc and Robnik-Šikonja (2025) used GPT-3.5 to extend Slovene dictionary entries: they took short examples from the Dictionary of Standard Slovenian Language (SSKJ) and asked GPT-3.5 to generate full-sentence contexts, improving data for sense disambiguation. Their pipeline showed that LLMs could augment lexicographic examples while preserving sense.

More broadly, Lew (2024) argues that traditional lexicography remains vital for languages like Slovene, especially given current LLM shortcomings in low-resource settings. Yet he also notes that AI tools offer new opportunities to automate routine tasks under a lexicographer's guidance. Our work builds on these efforts by instruction-tuning Slovene LLMs on curated dictionary resources. By combining multiple Slovene lexicographic datasets during fine-tuning, we aim to teach the model accurate, up-to-date lexical knowledge that reflects structured resources.

3. Data Collection and Preparation

We collected data for continued pre-training and fine-tuning to adapt a Slovene LLM for lexicographic tasks. We combine five complementary Slovene

lexicographic resources covering lexical, morphological, semantic, and usage-related information. The Digital Dictionary Database of Slovene (DDDS) serves as the core resource, providing structured lexical data for approximately 300,000 entries. Morphosyntactic annotations are drawn from the SUK corpus (Arhar Holdt et al., 2024). Definitions are supplemented by the Bridge Dictionary (CJVT, 2024) and OSWN (Čibej et al., 2023), while a synonym dictionary (Krek et al., 2023) provides sense-level synonym mappings.

3.1. Lexical Pretraining Corpus

Structured lexicographic entries were converted into natural language text to enable continued pretraining. The corpus is restricted to single-lexeme entries to ensure consistent template-based conversion.

Each entry lists all morphological forms from the DDDS, followed by word senses and definitions from SSKJ, Open Slovene Wordnet (OSWN), and the Bridge Dictionary, or semantic indicators where definitions are unavailable. Usage examples, collocations, and sense-organized synonyms are included where available.

Conversion was performed using rule-based templates that generate structured markdown with clear entry boundaries. The final corpus consists of a single markdown file containing 356,294 entries and approximately 875 MB of text.

An example of the pretraining corpus format is shown below (truncated):

```
# Opis besede vzmetnica
Beseda vzmetnicah je samostalnik, občno ime ž
     enskega spola množine v mestniku.
Beseda vzmetnico je samostalnik, občno ime ženskega
     spola ednine v orodniku.
...

## Definicije besede vzmetnica
Definicija besede vzmetnica je: Vzmetnica je velika
     ravna blazina,
ki jo damo v posteljo, da nam je udobneje.

## Kolokacije besede vzmetnica
- nova vzmetnica
- spati na vzmetnici
- posteljna vzmetnica
- zamenjati vzmetnico
...

## Sopomenke besede vzmetnica
- žimnica
- vzmet
```

3.2. QA Fine-Tuning Dataset

For conversational lexicographic QA, we constructed a dataset of 16,508 question–answer pairs from five sources (Table 1). The dataset combines automatically generated, expert-curated, and authentic user questions.

The largest subset contains 12,173 DDDS-derived QA pairs generated using templates, covering morphology, definitions, and common collocations. Additional data includes 112 questions generated based on patterns identified during a promptathon event in which we organized manual efforts to collect samples of lexicographic and linguistic questions that might be answered by such model. The dataset also includes 1,350 QA pairs derived from SUK annotations, 896 expert-answered questions from Jezikovna svetovalnica, a Slovene linguistic forum, and 342 questions from other Slovene online forums.

All data are stored in a ShareGPT JSON format with additional *data_type* and *source* fields.

An example QA item from the fine-tuning corpus is shown below:

```
{
    "conversations": [
        {
            "from": "human",
            "value": "Katere sopomenke ima beseda
     eksperimentalen?"
        },
        {
            "from": "gpt",
            "value": "Sopomenke za eksperimentalen
     so:\n- poskusen\n"
                     "- vzorčen\n- testen"
        }
    ],
    "source": "ddd",
    "data_type": "automatic"
}
```

Source	Description	Count
DDDS	Template-generated QA pairs	12,173
Promptathon	Community prompt patterns	112
SUK	POS-annotated examples	1,350
Svetovalnica	Linguistic advisory portal	896
Forums	Forum discussions	342
Total		**16,508**

Table 1: QA dataset composition by source

4. Evaluation Methodology

We conducted a comprehensive evaluation of our trained models by combining automated text-similarity metrics with LLM-based qualitative assessment. The evaluation compares the base GaMS model with our continued pretrained and fine-tuned variants across both domain-specific lexicographic questions and general-domain tasks.

4.1. Evaluation Datasets

Our primary evaluation was performed on a held-out test set from the lexicographic QA dataset, containing question-answer pairs with reference

(ground-truth) answers. The test split contains 10% of the full QA dataset and preserves the same question-source distribution as the training split. Each sample included:

- A prompt (question)
- A reference answer (correct answer)
- Generated responses from multiple model variants
- Source metadata indicating the origin of each question

Additionally, we evaluated model performance on general-domain tasks using the Slovenian-LLM-eval benchmark dataset, which consists of improved Slovenian translations of widely used English evaluation benchmarks, including ARC (Easy and Challenge), BoolQ, GSM8K, HellaSwag, NQ Open, OpenBookQA, PIQA, TriviaQA, TruthfulQA, and Winogrande, to assess whether domain specialization negatively impacted general language understanding capabilities.

4.2. Text Similarity Metrics

For lexicographic QA evaluation, we computed three complementary automated similarity metrics between each model's generated answer and the reference answer:

1. **Levenshtein Similarity**: Normalized edit distance measuring character-level differences, calculated as $1 - \frac{\text{distance}}{\max(\text{len}_1, \text{len}_2)}$
2. **Sequence Matcher Similarity**: Python's SequenceMatcher ratio measuring sequence alignment quality
3. **Word Overlap Similarity**: Jaccard similarity coefficient of word sets, computed as $\frac{|W_1 \cap W_2|}{|W_1 \cup W_2|}$

The final text similarity score was computed as the arithmetic mean of these three metrics, which were all normalized to provide a value between 0 and 1. This provided a robust measure of lexical and structural similarity to the reference answer:

$$\text{TextSim} = \frac{\text{Leven.} + \text{SeqMatcher} + \text{WOverlap}}{3}$$

Additionally, we computed BERTScore (Zhang et al., 2020), a learned metric that evaluates semantic similarity using contextual embeddings. BERTScore computes precision, recall, and F1 scores by matching tokens between candidate and reference texts using cosine similarity of BERT embeddings, providing a more nuanced semantic comparison than lexical overlap metrics.

These metrics served as simple reference metrics for quick comparison between the models as they are computationally efficient; however, for the main evaluation of the model performance, we used an LLM-as-a-judge approach, which judged the models closer to how a human would evaluate the response.

4.2.1. LLM-as-a-Judge Evaluation

To assess qualitative aspects beyond simple text matching, we employed GPT-5.2 as an LLM judge to perform comparisons of model answers. The use of an LLM judge enabled us to evaluate the models on a dataset with around 1000 examples and provided a solution for the lack of any standardized benchmarks for Slovene lexical quesiton answering. For each question, we evaluated model pairs across three dimensions:

1. **Semantic Similarity**: Which answer conveys meaning most similar to the reference answer, considering semantic equivalence rather than lexical matching
2. **Formatting Quality**: Which answer demonstrates superior structure, readability, and appropriate use of formatting elements (whitespace, bullet points, paragraphs, etc.)
3. **Grammatical Correctness**: Which answer exhibits the highest linguistic quality in Slovene, including proper spelling, syntax, and morphological case agreement

To mitigate position bias, where for example the model would prefere to select the first response, we randomized the order of model answers in each comparison. The GPT-5.2 judge was prompted to select the best answer and to respond only with the answer identifier. We set the temperature to 0.3 to balance consistency with nuanced evaluation capabilities. The model only had access to the reference answer in the semantic similarity setting, while the formatting quality and grammatical correctnes were evaluated independently of the reference answer.

While the use of an LLM judge might introduce a bias into the evaluation, we manually checked a small subset of model answers (30 answers from each model) and got results that coincided with the ones provided by the LLM judge.

4.3. Analysis and Aggregation

Results were aggregated at two levels:

- **Overall**: Average similarity scores and judge win counts across all test samples

- **By Source**: Separate aggregation for each data source (DDDS, SUK, Svetovalnica, Forumi, Promptathon) to identify domain-specific performance patterns

5. Results

We evaluated seven model variants to assess the impact of different training strategies on lexicographic question answering performance. Figure 1 shows how each of them was trained. For clarity, we introduce the following naming convention:

- **Base**: GaMS 9B with standard instruction tuning (cjvt/GaMS-9B-Instruct)
- **Base-Nemotron**: Base model retrained on translated Nemotron dataset (NVIDIA, 2024) (cjvt/GaMS-9B-Instruct-Nemotron)
- **Base-Nemotron-R**: Nemotron variant with reasoning prompting
- **Lex-FT**: Base model fine-tuned with lexicographic QA pairs
- **CPT-Lex-FT**: Model with continued pretraining on lexical corpus, then instruction tuned
- **GaMS3-Lex-FT**: Larger GaMS3 12B model fine-tuned with lexicographic data
- **GaMS3-CPT-Lex-FT**: GaMS3 12B model with continued pretraining on lexical corpus, then instruction tuned

5.1. Overall Performance

Table 2 presents the overall evaluation results across all test samples. The evaluation employs text-similarity scores (averaged from Levenshtein distance, sequence matching, and word overlap), BERTScore metrics (precision, recall, and F1), and LLM-as-a-judge assessments across three dimensions: semantic similarity, formatting quality, and grammatical correctness.

The lexicographic fine-tuning models substantially outperformed the baseline variants. The Lex-FT model achieved the highest text similarity (0.542), BERTScore F1 (0.915), and judge preference scores, representing a significant improvement over the Base model. Notably, the Base-Nemotron model performed poorly (0.091 in text similarity, 0.816 BERTScore F1), while excelling in the answer formatting quality dimension, likely because directly translating general-domain datasets can produce well-formatted responses that lack Slovene-specific knowledge. We also observed that the nemotron-trained model tends to provide very verbose responses that included substantial irrelevant information alongside the actual answer, reducing similarity scores.

The reasoning-prompted variant (Base-Nemotron-R) partially mitigated this verbosity issue, improving text similarity to 0.153, though still underperforming the lexicographic models. Among specialized models, the direct fine-tuning approach (Lex-FT) outperformed continued pretraining followed by fine-tuning (CPT-Lex-FT and GaMS3-CPT-Lex-FT), suggesting that for our dataset size and task, supervised fine-tuning provided more efficient knowledge transfer than the two-stage approach. The GaMS3-CPT-Lex-FT model showed comparable performance to GaMS3-Lex-FT, indicating that the additional continued pretraining stage did not provide substantial benefits for the larger model architecture.

5.2. Performance by Data Source

Figure 2 shows the model performance evaluated by semantic similarity to the reference answer, split by the source of examples. More detailed information from two representative data sources: SUK (sentence analysis questions) and Digital Dictionary Database of Slovene (lexical knowledge questions) is also presented in Table 3 and Table 4.

On SUK questions requiring sentence-level morphosyntactic analysis, lexicographic fine-tuning models achieved near-perfect text similarity scores (>0.95), with Lex-FT reaching 0.985. This represents a 202% improvement over the Base model. The baseline models completely failed on these structured analytical tasks (0 judge wins), while specialized models excelled.

For Digital Dictionary Database questions testing direct lexical knowledge, Lex-FT again led with 0.681 text similarity, though the absolute scores were lower than on the morphosyntactic analysis from SUK, suggesting greater task complexity and answer variability. The Base-Nemotron model catastrophically failed (0.028), while Base-Nemotron-R recovered somewhat (0.269), demonstrating the impact of prompting strategies on response quality.

Performance patterns remained consistent across different data sources. On Jezikovna Svetovalnica (language advisory portal) questions, all models achieved modest text similarity scores (0.08–0.11), with no clear winner, suggesting these authentic user questions present greater complexity with more varied valid answer formulations. The lexicographic models maintained competitive performance without dramatic improvement, indicating that human-generated linguistic advice may be harder to replicate than structured lexical knowledge.

On the promptathon questions, specialized models again dominated, with Lex-FT and CPT-Lex-FT

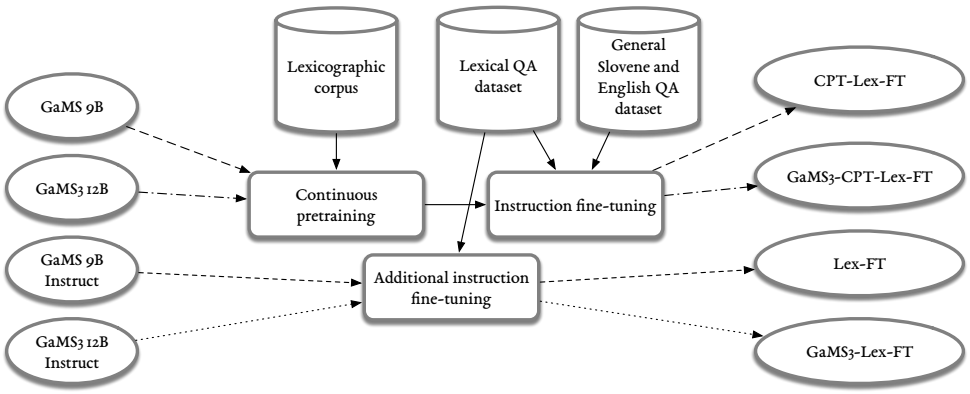


Figure 1: Overview of the training process showing the two-stage approach: continued pretraining on lexical corpus followed by instruction fine-tuning with QA pairs, applied to both GaMS 9B and GaMS3 12B base models.

Model	Text Sim.	BERT-P	BERT-R	BERT-F1	Semantic	Format	Grammar
Base	0.226	0.863	0.851	0.856	31	26	49
Base-Nemotron	0.091	0.792	0.844	0.816	16	**170**	13
Base-Nemotron-R	0.153	0.826	0.842	0.833	22	12	24
Lex-FT	**0.542**	**0.919**	**0.910**	**0.915**	**93**	34	**90**
CPT-Lex-FT	0.516	0.907	0.908	0.907	61	20	45
GaMS3-Lex-FT	0.497	0.905	0.900	0.902	33	10	38
GaMS3-CPT-Lex-FT	0.503	0.908	0.901	0.904	39	23	36

Table 2: Overall evaluation results. Text similarity and BERTScore metrics range 0–1 (higher is better). BERT-P/R/F1 denote BERTScore precision, recall, and F1. Judge metrics show number of wins in pairwise comparisons across all samples.

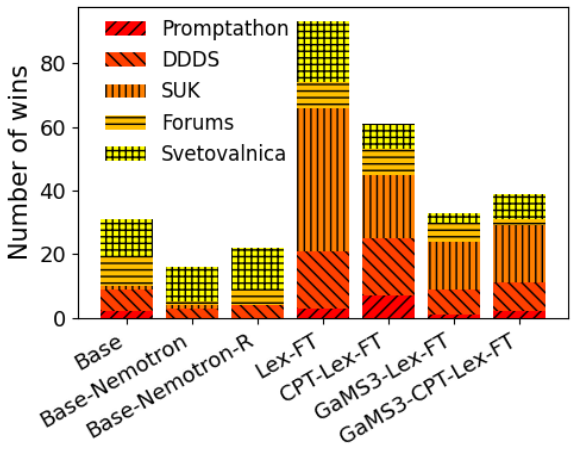


Figure 2: LLM-as-a-judge evaluation results showing the number of wins each model achieved across different data sources in pairwise comparisons for semantic similarity to the reference answer.

Model	Text Sim.	Semantic
Base	0.326	0
Base-Nemotron	0.142	0
Base-Nemotron-R	0.161	0
Lex-FT	**0.985**	**54**
CPT-Lex-FT	0.959	21
GaMS3-Lex-FT	0.953	25

Table 3: SUK sentence analysis results (subset of judge metrics shown for brevity).

Model	Text Sim.	Semantic
Base	0.197	10
Base-Nemotron	0.028	1
Base-Nemotron-R	0.269	8
Lex-FT	**0.681**	**40**
CPT-Lex-FT	0.571	23
GaMS3-Lex-FT	0.635	18

Table 4: Digital Dictionary Database lexical knowledge results.

achieving 0.86 text similarity compared to 0.46 for Base. The Slovene forums' questions showed less pronounced differentiation, with Lex-FT reaching 0.236 compared with 0.136 for Base, likely reflecting the informal, discussion-oriented nature of forum answers.

5.3. Qualitative Error Analysis

Beyond quantitative metrics, a qualitative examination of model outputs reveals characteristic error patterns that shows the impact of training data qual-

ity and domain adaptation. We present representative examples of common failure modes across model variants.

5.3.1. Language Confusion in Translation-Based Models

The most common failure pattern occurs in models trained on automatically translated data. The Base-Nemotron model frequently confuses English and Slovene linguistic concepts, producing responses that discuss English grammar when answering questions about Slovene. For example, when asked whether the Slovene phrase "kolikor" can appear at the beginning of a sentence, the Base-Nemotron model responded (in Slovene): "Yes, the phrase kolikor can stand at the beginning of a sentence, but in formal or literary English this is not common. It is more commonly used in more informal or colloquial language or in specific grammatical structures."

This error demonstrates that automatic translation of instruction-tuning datasets can introduce fundamental confusion about the target language's identity. The model correctly understands the question structure but applies knowledge about English grammar to answer a question explicitly about Slovene. The lexicographic fine-tuning eliminated this problem, maintaining proper language context throughout their responses.

5.3.2. Repetition and Degeneration

Another characteristic failure involves repetitive generation, particularly when models attempt to provide information they lack. When asked to list synonyms for the Slovene verb "priznati" (to admit/acknowledge), which has no clear synonyms, the Base-Nemotron model produced an extensively structured but contentless response that repeatedly listed "priznati" itself as its own synonym across multiple categories.

This pattern of repetitive degeneration appears when the model attempts to conform to an expected response structure (categorized synonym lists with examples) without possessing the necessary lexical knowledge. The response demonstrates superficial understanding of the task format but complete failure to provide useful content.

In contrast, the lexicographic fine-tuned models produced more appropriate responses, by providing semantically related forms like "priznati se" (reflexive form) or "priznavati" (imperfective aspect), which represent morphological variants rather than true synonyms. While these responses are not perfect, they demonstrate substantially better understanding of Slovene lexical structure.

5.3.3. Cross-Linguistic Interference in Specialized Terminology

A third error pattern involves providing information about the wrong language's linguistic system. When asked to list examples of Slovene auxiliary verbs ("pomožni glagoli"), both Base-Nemotron models provided extensive lists of *English* auxiliary verbs in addition to the Slovene verbs.

This failure is particularly problematic because the question asks about Slovene verbs, yet the model responds mostly in terms of English grammar. While the lexicographic models maintained proper language context by providing Slovene terms such as "biti" (to be), "imeti" (to have), "morati" (must), and "smeti" (may), they also produced incorrect answers: in standard Slovene grammar, only "biti" is classified as an auxiliary verb. This reveals a limitation of purely fine-tuning-based approaches, even with domain-specific training, models may generate plausible but factually incorrect information when lacking access to authoritative structured resources.

These error patterns collectively demonstrate that automatic translation of general-domain training data introduces systematic language confusion that persists even in models with strong general capabilities. Domain-specific fine-tuning on native Slovene lexicographic data substantially improves language alignment and eliminates cross-linguistic interference, but does not guarantee factual accuracy on all specialized queries. This limitation points to the need for retrieval-augmented generation systems that can directly query authoritative lexicographic databases to ensure correct, verifiable answers, particularly for questions with precise, definitional answers.

5.4. Key Findings

Our evaluation reveals several key insights. (1) Lexicographic models outperformed baselines, with even larger gains on structured tasks like sentence analysis. (2) The Base-Nemotron model's poor results (0.091 text similarity, 0.816 BERTScore F1) demonstrate that automatically translated general-domain datasets may introduce artifacts problematic for specialized tasks. (3) Contrary to our initial hypothesis, the Lex-FT model (direct fine-tuning) consistently outperformed CPT-Lex-FT and GaMS3-CPT-Lex-FT (continued pretraining + fine-tuning), suggesting that for datasets of our size (15K examples), supervised learning on task-specific data provides more efficient adaptation than unsupervised continued pretraining. (4) The GaMS3-Lex-FT and GaMS3-CPT-Lex-FT models (12B parameters) did not substantially outperform the 9B Lex-FT model, indicating that training data quality and task alignment matter more than

raw parameter count for specialized domains. (5) Models excelled on structured tasks (SUK: 0.985) but showed modest improvements on open-ended linguistic advice (Svetovalnica: 0.11), highlighting the importance of answer structure and definedness in evaluation outcomes.

5.5. General Benchmark Evaluation

To assess whether our domain-specific training negatively impacted general language understanding capabilities, we evaluated our models on the Slovenian LLM Evaluation Dataset, which contains Slovenian translations of widely used English benchmarks. This dataset, developed by Arčon et al. (2024), includes improved translations of popular benchmarks such as ARC (Easy and Challenge), BoolQ, HellaSwag, NQ Open, OpenBookQA, PIQA, TriviaQA, and Winogrande.

Table 5 presents the results for selected models and benchmarks. The full results across all benchmarks are shown in Figure 3.

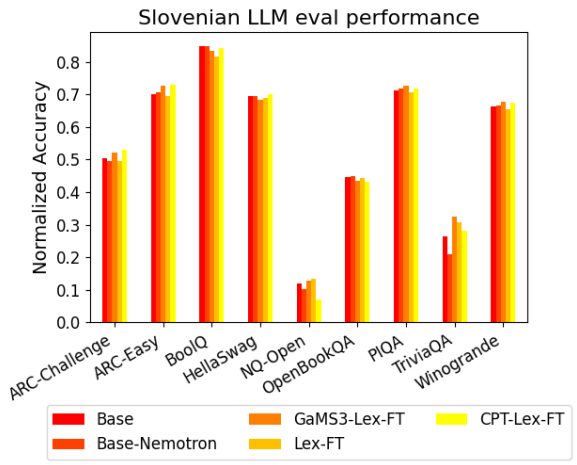


Figure 3: Comprehensive benchmark evaluation results across all models and evaluation tasks, showing accuracy scores on the Slovenian LLM Evaluation Dataset benchmarks.

The results demonstrate that our lexicographic fine-tuning had minimal negative impact on general language understanding. The CPT-Lex-FT models (which underwent continued pretraining followed by instruction fine-tuning) maintained or even slightly improved performance on most benchmarks compared to the Base model (0.756 on ARC-Easy, 0.842 on BoolQ, 0.700 on HellaSwag).

The direct fine-tuning models (Lex-FT) showed a modest decrease in some benchmarks, particularly on ARC Challenge and BoolQ, suggesting a slight trade-off between domain specialization and general capabilities. However, the differences remain relatively small (typically 2-5 percentage points), indicating that the models retained strong general language understanding while gaining substantial improvements in lexicographic tasks.

Notably, the GaMS3-Lex-FT model (based on the larger 12B parameter architecture) maintained competitive performance across all benchmarks, demonstrating that the larger model capacity helped preserve general capabilities despite domain-specific training.

6. Discussion

Our results demonstrate the effectiveness of domain-specific fine-tuning for lexicographic question answering. Gains were especially pronounced on structured tasks (SUK: 0.985), while performance on open-ended questions from Jezikovna Svetovalnica remained modest, suggesting that expert linguistic advice is harder to replicate with limited supervised data.

Contrary to expectations, direct fine-tuning (Lex-FT) outperformed the two-stage approach (CPT-Lex-FT), suggesting that for moderate-sized datasets, supervised learning provides more efficient adaptation than multi-stage training. Similarly, increasing model size from 9B to 12B parameters yielded minimal gains, indicating that data quality and task alignment matter more than parameter count.

The Base-Nemotron model's failure highlights critical risks of automatically translated training data, which conflates English and Slovene linguistic concepts. While domain-specific fine-tuning eliminates cross-linguistic interference, it does not guarantee factual accuracy, motivating retrieval-augmented generation approaches.

Importantly, domain specialization did not harm general language capabilities, with only modest decreases (2-5 percentage points) on general benchmarks. Our findings offer a practical strategy for low-resource languages: existing linguistic resources can be effectively repurposed, direct fine-tuning often suffices, and moderately sized models are sufficient.

7. Conclusion

In this work, we investigated domain-specific adaptation of large language models for Slovene lexicographic question answering. We proposed a two-stage training approach combining continued pretraining on a large-scale lexical corpus with instruction fine-tuning on curated question-answer pairs. To support this process, we constructed a comprehensive lexical pretraining corpus containing 356,294 word entries and a dedicated instruction dataset of 16,508 lexicographic QA pairs drawn from both structured resources and authentic user queries.

Model	ARC-C	ARC-E	BoolQ	HellaSwag	PIQA
Base	0.515	0.732	0.848	0.693	0.702
Base-Nemotron	0.506	0.739	0.847	0.696	0.705
GaMS3-Lex-FT	0.500	0.740	0.832	0.683	0.702
Lex-FT	0.469	0.711	0.805	0.671	0.690
CPT-Lex-FT	0.515	0.756	0.842	0.700	0.707

Table 5: General benchmark evaluation results (accuracy) for selected models on key benchmarks. ARC-C = ARC Challenge (normalized accuracy), ARC-E = ARC Easy (accuracy), BoolQ = Boolean Question accuracy, HellaSwag = commonsense inference (normalized accuracy), PIQA = physical commonsense reasoning (accuracy).

Our experimental results show that domain-specific fine-tuning substantially improves model performance on lexicographic tasks. In particular, direct instruction fine-tuning of the GaMS model led to large gains over the baseline across multiple evaluation settings, demonstrating that targeted supervision effectively transfers structured lexicographic knowledge into a conversational QA format. While continued pretraining did not provide additional benefits in our setting, the overall approach successfully enhanced the model's ability to deliver accurate, well-structured lexicographic answers without degrading general language understanding.

This work demonstrates that high-quality, native-language linguistic resources can be efficiently repurposed to build specialized language models for low-resource languages. Our findings highlight that careful data curation and task-aligned fine-tuning can outweigh increased model size or complex training pipelines, offering a practical pathway for developing domain-specific language technologies in smaller language communities.

8. Acknowledgements

This work was financially supported by the Slovenian Research and Innovation Agency through the research project Large Language Models for Digital Humanities (GC-0002). This work is also co-funded by the European Union HORIZON-WIDERA-2023-TALENTS-01-01 grant 101186647 AI4DH. We thank Luka Dragar for assistance with data preparation and Domen Vreš for providing access to the latest GaMS models and help with their training.

9. Bibliographical References

Tjaša Arčon, Timotej Petrič, and Domen Vreš. 2024. Slovenian llm evaluation dataset. Hugging Face.

CJVT. 2024. GaMS: Generative model for slovene.

CJVT. 2026. Gams3-12b-instruct. Hugging Face.

Aleš Horák and Adam Rambousek. 2017. Lexicography and natural language processing. In *The Routledge handbook of lexicography*, pages 179–196. Routledge.

Robert Lew. 2023. ChatGPT as a COBUILD lexicographer. *Humanities and Social Sciences Communications*, 10(1):1–10.

Robert Lew. 2024. Dictionaries and lexicography in the ai era. *Humanities and Social Sciences Communications*, 11(1):1–8.

Raphaël Merx, Ekaterina Vylomova, and Kemal Kurniawan. 2024. Generating bilingual example sentences with large language models as lexicography assistants. In *Proceedings of the 22nd Annual Workshop of the Australasian Language Technology Association*, pages 64–74.

Viktor Moskvoretskii, Ekaterina Neminova, Alina Lobanova, Alexander Panchenko, and Irina Nikishina. 2024. TaxoLLaMA: WordNet-based model for solving multiple lexical semantic tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2331–2350.

NVIDIA. 2024. Nemotron post-training dataset v1. Hugging Face.

Bach Pham, JuiHsuan Wong, Samuel Kim, Yunting Yin, and Steven Skiena. 2025. Word definitions from large language models. In *2025 19th International Conference on Semantic Computing (ICSC)*, pages 158–162. IEEE Computer Society.

Tadej Škvorc and Marko Robnik-Šikonja. 2025. Solving word-sense disambiguation and word-sense induction with dictionary examples. *arXiv preprint arXiv:2503.04328*.

Domen Vreš, Martin Božič, Aljaž Potočnik, Tomaž Martinčič, and Marko Robnik-Šikonja. 2024. Generative model for less-resourced language with 1 billion parameters. *arXiv preprint arXiv:2410.06898*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

10. Language Resource References

Arhar Holdt, Špela and Krek, Simon and Dobrovoljc, Kaja and Erjavec, Tomaž and Gantar, Polona and Čibej, Jaka and Pori, Eva and Terčon, Luka and Munda, Tina and Žitnik, Slavko and Robida, Nejc and Blagus, Neli and Može, Sara and Ledinek, Nina and Holz, Nanika and Zupan, Katja and Kuzman, Taja and Kavčič, Teja and Škrjanec, Iza and Marko, Dafne and Jezeršek, Lucija and Zajc, Anja. 2024. *Training corpus SUK 1.1*. Slovenian language resource repository CLARIN.SI.

Jaka Čibej, Luka Terčon, Simon Krek, Andraž Repar, Erik Novak, Polona Gantar, Iztok Kosem, Špela Arhar Holdt, Kaja Dobrovoljc, Amadea Berginc, Irena Hvala, Damijan Klement, Manja Kolenc, Ana Močnik, Tina Munda, David Pavlas, Anamari Pečan, Aleksandra Poljak, Davorin Sečnik, Jure Šešet, Jan Štumberger, Tina Toličič, and Laura Trpin. 2023. Open slovene WordNet OSWN 1.0. Slovenian language resource repository CLARIN.SI.

CJVT. 2024. *Bridge Dictionary of Slovene*. Centre for Language Resources and Technologies. CJVT Language Resources.

DDDS. List of linked senses from open slovene wordnet and the digital dictionary database of slovene 1.0. https://b2find.eudat.eu/dataset/5a2c5642-240c-5a27-8aad-a7896d7ceb77. Dataset - B2FIND.

Kaja Dobrovoljc, Simon Krek, Peter Holozan, Tomaž Erjavec, Miro Romih, Špela Arhar Holdt, Jaka Čibej, Luka Krsnik, and Marko Robnik-Šikonja. 2019. Morphological lexicon sloleks 2.0. Slovenian language resource repository CLARIN.SI.

Iztok Kosem, Špela Arhar Holdt, Simon Krek, Polona Gantar, Eva Pori, Jaka Čibej, Bojan Klemenc, Cyprian Laskowski, Kaja Dobrovoljc, Vojko Gorjanc, Nikola Ljubešić, Karolina Zgaga, and Rebeka Roblek. 2023. Collocations dictionary of modern slovene KSSS 2.0. Slovenian language resource repository CLARIN.SI.

Simon Krek, Cyprian Laskowski, Marko Robnik-Šikonja, Iztok Kosem, Špela Arhar Holdt, Polona Gantar, Jaka Čibej, Vojko Gorjanc, Bojan Klemenc, Kaja Dobrovoljc, Eva Pori, Rebeka Roblek, and Karolina Zgaga. 2023. Thesaurus of modern slovene 2.0. Slovenian language resource repository CLARIN.SI.

Structured Partial Predictability in Non-Concatenative Morphology: The Case of Tashlhiyt Berber

John Alderete, Hamza Sellami
Simon Fraser University, Canada & Mohammed V University, Rabat, Morocco
alderete@sfu.ca, hamza-sellami@um5r.ac.ma

Abstract

Non-concatenative morphology poses a persistent challenge for NLP, yet structured quantitative resources for Amazigh (Berber) languages remain scarce. We present the first large-scale computational study of Tashlhiyt Berber plural formation, drawing on a richly annotated dataset of 1,185 noun paradigms with phonological, morphological and semantic features. We decompose the plural system into macro-level word-formation strategies and micro-level stem mutations, and evaluate predictability across ten target domains using linguistic feature models, N-gram baselines, and Bi-LSTM neural models. Results reveal a structured split: linguistic features decisively outperform neural models on systematic macro-level strategies (e.g., +44.5pp F_1), while Bi-LSTMs better capture lexically idiosyncratic patterns. Rather than supporting a categorical rule/memory divide, this complementarity reveals gradient layers of regularity within a single morphological system. These findings demonstrate the value of linguistically informed annotation for probing morphological complexity in low-resource, typologically diverse languages. All data, code, and models are publicly available.

Keywords: non-concatenative morphology, Tashlhiyt Berber, Amazigh languages, structured linguistic annotation, morphological predictability, low-resource languages, grammar-memory interface

1. Introduction

While morphological models perform well on concatenative systems, non-concatenative morphology, which requires encoding grammar through internal stem modification, remains a major NLP challenge. Afroasiatic morphology utilizes complex vocalic ablaut, gemination, and templatic processes (McCarthy, 1979). Despite neural dominance in benchmarks (Cotterell et al., 2016; Vylomova et al., 2020), black-box architectures often obscure the grammar-memory interface: the degree to which forms are productively computed versus lexically stored (Kirov and Cotterell, 2018). Quantifying this predictability is essential for optimizing model architectures and probing morphological complexity in typologically diverse languages (Cotterell et al., 2019).

Amazigh languages offer a natural testing ground for these questions, and computational linguistics for these languages has seen meaningful progress over the past two decades, driven in part by IRCAM's standardization efforts[1] and a growing ecosystem of NLP resources and tools. This includes work on computer-assisted linguistic analysis (Jebbour, 1996; Taghbalout et al., 2015), automatic language detection (Adouane et al., 2016b,a; Lafkioui, 2008), machine translation (Taghbalout et al., 2015), annotated dataset development (Amri et al., 2017), morphological analysis tools (Raiss and Cavalli-Sforza, 2012), and deep learning approaches to sequence labeling tasks such as POS tagging (Bani et al., 2023).

Nevertheless, while research on Arabic has quantified partial predictability and templatic effects in plural formation (Dawdy-Hesterberg and Pierrehumbert, 2014), no comparable analysis exists for Amazigh varieties. Specifically, the question of where rule-governed grammar ends and lexically idiosyncratic memory begins has not been computationally investigated. Tashlhiyt, an Amazigh (Berber) language spoken by 7-8 million people in southern Morocco, provides an ideal test case, combining external suffixation and internal mutation in its plural system. Bridging this gap is critical for developing low-resource morphological tools (Khalifa et al., 2020) and for testing whether Afroasiatic findings generalize or reflect language-specific idiosyncrasies (cf. McCurdy et al. (2020) on parallel questions in German).

We address this complexity via a multi-layered framework, decomposing plurals into macro-level strategies (i.e., suffixation vs. stem mutations) and micro-level stem mutations (e.g., vowel ablaut and insertion), and evaluating predictability across ten target domains using linguistic feature models, N-gram baselines, and Bi-LSTM neural models.[2] We find a structured split: hand-crafted features decisively outperform Bi-LSTM baselines on macro-level tasks (e.g., +44pp F_1 on 3-way classification), while the Bi-LSTM better captures specific micro-level domains where explicit structural generalizations fail. Rather than reflecting a categor-

[1] https://tal.ircam.ma/talam/

[2] All data, code, and trained models are available at: github.com/aldo-git-bit/predicting-tashlhiyt-plural.

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 124–135

ical rule/memory divide, this complementarity reveals gradient layers of regularity within a single morphological system — consistent with accounts that treat morphological productivity as gradient rather than categorical (Bybee and McClelland, 2005; Hay and Baayen, 2005; O'Donnell, 2015). High irreducible error alongside low idiosyncrasy severity further suggests that complexity stems from competition among overlapping regularities rather than arbitrary lexical listing. These results demonstrate that structured linguistic annotation provides a diagnostic tool for probing morphological complexity, with implications for hybrid modeling in low-resource, non-concatenative systems.

2. Related Research

Prior work on Amazigh plurals distinguishes three classes: external (primarily /-n/ suffixation), internal (vowel changes and insertion), and mixed (Basset, 1952; Saib, 1986; Jebbour, 1988; Lasri, 1991; Taine-Cheikh, 2006; Hasselbach, 2007; Ben Si Saïd, 2014, 2020). Several analyses argue that surface markers such as vowel insertion are phonological consequences of affixation rather than independent morphemes (Jebbour, 1996; Lasri, 1991), and Idrissi (2000) proposes that prosodic structure conditions plural class selection. Plural assignment is generally characterized as partially systematic, shaped by the interaction of prosodic and morphological factors with lexical idiosyncrasy, but no prior work has quantitatively evaluated the predictability of plural class from singular forms. We address this gap computationally.

3. Dataset

3.1. Data Source

We use 1,185 Tashlhiyt noun paradigms from (Alderete et al., 2025), which draws on and integrates noun paradigms from a comprehensive grammatical study (Jebbour, 1996) and a Tashlhiyt text corpus (Jebbour et al., 2021). Nouns follow a hierarchical schema: three macro-classes—External (primarily suffixation of /-n/), Internal (stem-internal vowel changes such as ablaut and vowel insertion), and Mixed (combining suffixation with stem modifications)—and specific micro-level patterns (Table 1)[3]. Predictive features cover morphology (gender, derivation, paradigm structure, stem augments, and loan status) and semantics (animacy, humanness, and a 22-category semantic field based on Buck (2008)). Full feature labels and vocabulary sizes are provided in Appendix A.

Consider the following concrete examples to illustrate the problem we are investigating. The singular nouns for 'speech' and 'finger' in Table 1 have the same consonant-vowel pattern in their stems, /wal/ and /dˤadˤ/, respectively. Thus, while the inputs to plural formation both have a CVC pattern, they have two very different outcomes: /wal/ receives a final vowel-glide sequence in *i-waliw-n*, while /dˤadˤ/ undergoes both vowel substitution and final vowel insertion in *i-dˤudˤa-n*. These micro-level differences are compounded by macro-level divergences in word-formation strategy. The CCVC stem /snus/ exhibits stem-internal ablaut: *i-snas* 'donkey', but the comparable stem /fraw/ for 'sheet' has no internal changes and instead receives a simple suffix in the plural: *i-fraw-n*. Our goal is to computationally quantify the factors that govern these differences, distinguishing rule-governed regularities from lexical idiosyncrasy.

Table 1: Plural formation patterns in Tashlhiyt

Pattern	n **(%)**	**Example (Sg. → Pl.)**
Macro-patterns (1,185 total)		
External	623 (52.8%)	*a-ʃddir* → *i-ʃddir-n* 'bush'
Internal	357 (30.3%)	*a-sds* → *i-sdas* 'trough'
Mixed	200 (17.0%)	*a-dmr* → *i-dmar-n* 'chest'
Micro-patterns (Only Internal and Mixed, 562 total)		
Medial A	63 (11.3%)	*a-sgdl* → *i-sgdal* 'block'
Final A	57 (10.2%)	*a-llun* → *i-lluna* 'tambourine'
Final Vw	27 (4.9%)	*a-wal* → *i-waliw-n* 'speech'
Final T	38 (6.8%)	*a-qaʃʃa* → *i-qaʃʃat-n* 'crest'
Ablaut	226 (40.6%)	*a-snus* → *i-snas* 'donkey'
Abl+Med.A	30 (5.4%)	*a-satm* → *i-sutam* 'niche'
Abl+Fin.A	14 (2.5%)	*a-dˤadˤ* → *i-dˤudˤa-n* 'finger'
Templatic	102 (18.3%)	*lkbbutˤ* → *lkbabtˤ* 'coat'

3.2. Phonological N-grams

To provide hand-crafted features and a baseline, we extracted edge-aligned phonological n-grams (i.e., contiguous sequences of 1–3 IPA segments anchored to word boundaries) from singular stems, yielding 2,265 macro-level and 1,356 micro-level features. We reduced dimensionality using LASSO with Stability Selection (100 bootstrap iterations, threshold ≥ 0.50) to define a static feature space of 2,019 and 1,149 features, respectively. Critically, while this procedure characterized the feature space on the full dataset, all model weights were learned strictly within training folds during 10-fold cross-validation to prevent data leakage (see Appendix B for details).

Per-task feature selection isolated informative n-gram subsets ranging from broad coverage for complex Ablaut (1,106 features) to compact signatures for Templatic patterns (92 features). This

[3] Ablaut = vowel quality change in the stem (e.g., /u/ → *a*); Medial A = insertion of *a* in a stem-medial position; Final A = insertion of *a* in final position; Final T and Final Vw = insertions of *t* and vowel+glide; Abl.Med.A denotes the cooccurrence of Ablaut and Medial A; likewise Abl.Fin.A is Ablaut + Final A; Templatic = the imposition of a fixed CV pattern on the stem

optimization filtered noise and prioritized informative, rare n-grams, significantly improving generalization; for example, F_1 for templatic mutations rose from 0.344 to 0.718 (a gain of 37.4pp).

3.3. Prosodic Features

We syllabified stems using the sonority-based system of Dell and Elmedlaoui (2002), parsing results into L/H syllables and metrical feet. To reduce dimensionality by 70% and enhance interpretability, raw strings were replaced with nine theory-driven features. For instance, `p_LH_ends_L` (final light syllables) identifies stems 7.22$\times$ more likely to undergo Medial A mutation ($+23.6$% effect), confirming that insertion satisfies prosodic weight requirements. Other features quantify moras, feet, and metrical residue. This setup ensures each mutation type is associated with $\geq$3 significant predictors ($p < 0.05$).

4. Computational Experiments

4.1. Multi-Level Prediction and Ablation

We implement a multi-level framework comprising ten independent experiments to address morphological complexity. Pluralization is decomposed into two tiers: a **macro-level** ($N = 1,185$) targeting broad strategies and a **micro-level** ($N = 562$) targeting specific mutation patterns (e.g., Ablaut, Templatic). This tiered approach isolates distinct dimensions of change, allowing for tailored feature selection and a granular mapping of the grammar-memory interface. Systematic ablation of morphological, semantic, and phonological subsets identifies core predictors and evaluates feature redundancy across all domains. These studies quantify the specific linguistic drivers of pluralization while facilitating critical baseline comparisons, such as probing the predictive value of abstract prosodic features over surface n-grams.

4.2. Experimental Models

To prioritize interpretability, we utilize three architectures: Logistic Regression (L_2-regularized), Random Forest, and Gradient Boosting (XGBoost). Our protocol emphasizes linguistic transparency over metric maximization; we therefore eschew exhaustive hyperparameter tuning in favor of fixed, conservative configurations across all domains (e.g., Logistic Regression $C = 1.0$; see Appendix B). By maintaining identical parameters across feature subsets, observed performance deltas reflect the informational quality of hand-crafted features rather than optimization artifacts.[4] This framework provides a consistent baseline for quantifying the relative contributions of morphological, phonological, and semantic features to plural formation.

4.3. Baseline Models

We utilize two baselines. An N-gram baseline uses Logistic Regression on edge-aligned phoneme sequences as a distributional superset. A bidirectional LSTM (Bi-LSTM) provides a neural ceiling for models trained from scratch at this data scale, capturing dependencies from raw characters without linguistic guidance. This architecture is deliberately matched to the data regime. Our task, that of classifying plural strategy from isolated singular forms, differs fundamentally from the seq2seq morphological inflection benchmarks like the SIGMORPHON shared tasks (Cotterell et al., 2016; Vylomova et al., 2020), which provide explicit input–output pairs and benefit from character-level copying. With 1,185 paradigms for 3- or 8-way classification, training from scratch is the only regime that avoids confounding the comparison with external pre-training data. Furthermore, we favor character-level probes because powerful pretrained models like mBERT lack sufficient Amazigh representation. In sum, these two baselines isolate the gain of our features: N-grams quantify the value of prosodic abstractions over surface phonotactics, while the Bi-LSTM identifies the idiosyncrasy gap that remains irreducible through sequence learning alone.

The Bi-LSTM was trained from scratch on the same 1,185 paradigms. Each singular form was represented as a sequence of IPA characters mapped to 32-dimensional embeddings. The model comprised a single bidirectional LSTM layer (64 hidden units per direction) with a fully connected classification head, trained with Adam, early stopping (patience = 10), and dropout of 0.3.

4.4. Data Imbalance and Over-fitting

To mitigate overfitting, we employ stratified 10-fold cross-validation. We address significant class imbalance (up to 19.8:1) through a combination of class weighting and SMOTE (Synthetic Minority Oversampling Technique; Chawla et al. (2002)) applied to training folds. Our primary evaluation metric is Macro-F_1, applied **uniformly** across all binary and multiclass domains. By equalizing class contributions—averaging F_1 scores across both categories in binary tasks—we ensure that performance on rare mutation patterns is not obscured by majority-class heuristics. This provides a more robust assessment of model quality than accuracy or positive-class-only metrics.

[4] A sensitivity check across $C \in 0.1, 1.0, 10.0$ confirmed that feature-set rankings were preserved in 9/10 domains at all three values (mean Macro-F_1 variation: 0.03), indicating that observed performance deltas reflect feature informativeness rather than the specific choice of C=1.0.

4.5. Residual Lexical Idiosyncrasy

To investigate the grammar-memory tradeoff, we quantify residual lexical idiosyncrasy, or the proportion of plural assignment resisting systematic prediction. The **computational ceiling** is the maximum performance of either model, representing a relative upper bound constrained by our architectures rather than an absolute human limit. **Irreducible error** is the percentage of forms where both models fail, while the **learnable proportion** marks the upper bound of predictable variance. To evaluate synergy between distributional and featural learning, we measure **model complementarity** as the ratio of non-overlapping errors to the learnable proportion. Finally, **idiosyncrasy severity** isolates high-confidence joint failures (forms where both models assign probability ≥ 0.70 to their predicted class yet misclassify), distinguishing learnable regularities from the irreducible, arbitrary core of the system.

5. Results

Within the hierarchical decomposition of the plural system, our results demonstrate an architecture characterized by structured partial predictability, where the divide between grammatical rules and lexical memory is clearly reflected in model performance across different tiers of analysis.

5.1. Macro-Level Systematicity

Macro-level pluralization is highly systematic and governed by rule-based dependencies. Hand-crafted features decisively outperform neural sequence learning, exceeding the Bi-LSTM by $+0.445$ in 3-way classification and $+0.378$ for stem mutations (Table 2). Paired t-tests confirm M+P significantly outperforms the Bi-LSTM across all macro-level tasks ($p < 0.001$). This performance gap suggests that macro-level morphological choices rely on abstract categories like gender or phonological patterns. While interpretable features effectively capture these rules, our Bi-LSTM baseline, which is trained from scratch on 1,185 paradigms, does not induce the abstract regularities that hand-crafted features encode directly.[5] This mirrors findings in Arabic, where coarse-grained phonological representations capture generalizations that surface distributional patterns miss (Dawdy-Hesterberg and Pierrehumbert, 2014).

[5]The only published Bi-LSTM result for Amazigh (Bani et al. (2023): 97% accuracy) addresses POS tagging on 60k tokens of running text with rich sentential context — a fundamentally different task from predicting plural strategy class from isolated singular forms. No prior work applies neural models to Tashlhiyt plural strategy classification, and no Amazigh variety appears in the SIGMORPHON shared tasks (2016–2024).

5.2. Phonological Primacy

Phonological features emerge as the primary predictive driver across nearly all domains. Although the combined M+P model often achieves the highest Macro-F_1, morphological features provide only marginal refinements to a predominantly phonological signal. While the N-gram baseline is competitive, hand-crafted features provide a consistent boost, particularly in multiclass tasks like 3-way macro-classification ($\Delta\text{Ngr} = +0.068$). By encoding explicit prosodic constraints, these features demonstrate that Tashlhiyt pluralization is sensitive to metrical structure rather than surface phonotactics alone, suggesting that morphological modeling for low-resource templatic languages should consider engineering prosodic abstractions.

5.3. Micro-Level Variation

At the micro-level, performance is more balanced between grammar and memory. Systematic patterns like Templatic and Ablaut plurals demonstrate high predictability (*Ceil* $= 0.907$ and 0.761, respectively), mirroring macro-level systematicity. In imbalanced domains (e.g., Medial A, Final A), hand-crafted features effectively capture rare mutations when evaluated under Macro-F_1 weighting. Three micro-level tasks (Medial A, Ablaut, Templatic) show no significant difference between M+P and the Bi-LSTM, suggesting these patterns are equally learnable via explicit prosodic features or distributional sequence learning. Ultimately, specific phonological patterns, rather than data volume, appear to be the primary driver of micro-level learnability.

5.4. Quantifying the Idiosyncrasy Gap

Our residual analysis provides a conservative lower bound for the memory component of the system. While Irreducible Error remains high in multiclass tasks (27.7% for 8-way mutation), idiosyncrasy severity—the proportion of high-confidence misclassifications—remains below 5% in most domains. This suggests that Tashlhiyt plural formation is not dominated by true lexical exceptions. Instead, the "unpredictability" often cited in traditional grammars likely reflects model uncertainty between competing, phonologically plausible patterns rather than arbitrary lexical listing. The high complementarity observed in domains like Ablaut (0.520) further suggests that distributional memory and grammatical rules provide non-redundant signals, supporting the use of hybrid architectures for non-concatenative morphological analysis.

		Hand-Crafted Feature Models					Baseline Comparisons				Residual Analysis			
	Domain	Sem	Morph	Phon	M+P	All	Ngr	LSTM	ΔNgr	ΔLSTM	Ceil	IrrErr%	Compl	Sev%
Macro	Has Suffix	0.500	0.541	0.785	**0.797**	0.781	0.760	0.735	0.036	0.062	0.797	12.2	0.421	3.8
	Has Mutation	0.566	0.567	0.771	**0.777**	0.769	0.754	0.399	0.023	0.378	0.777	8.6	0.524	0.0
	3-way	0.402	0.374	0.658	**0.683**	0.666	0.614	0.238	0.068	0.445	0.683	17.6	0.462	0.0
Micro	Medial A	0.495	0.503	0.670	**0.715**	0.688	0.678	0.673	0.037	0.042	0.715	7.3	0.324	2.7
	Final A	0.464	0.476	0.457	0.593	**0.617**	0.660	0.458	−0.067	0.135	0.593	7.4	0.260	3.3
	Final Vw	0.522	0.550	**0.678**	0.645	0.671	0.609	0.519	0.037	0.126	0.645	2.2	0.204	0.5
	Ablaut	0.582	0.659	**0.775**	0.761	0.748	0.739	0.736	0.021	0.025	0.761	10.9	0.520	2.7
	Final T	0.480	0.515	0.570	0.636	**0.648**	0.663	0.512	−0.027	0.124	0.636	4.9	0.245	0.8
	Templatic	0.605	0.788	0.889	0.907	**0.910**	0.872	0.848	0.034	0.059	0.907	0.5	0.025	0.3
	8-way	0.172	0.192	0.439	**0.469**	0.464	0.418	0.086	0.051	0.383	0.469	27.7	0.500	0.0

Table 2: Plural prediction and residual analysis (10-fold CV) in Macro-F_1 or %; M+P is the reference for all Δ and residuals. Baselines: Ngr (N-gram); LSTM (Bi-LSTM); Δ (M+P $-$ baseline). Residuals: Ceil (max F_1); IrrErr (joint failure); Compl (Complementarity); Sev (high-confidence exceptions). Significance: Appendix Table 14.

6. Discussion

6.1. Main Findings

Tashlhiyt pluralization follows a tiered dependency structure rather than a monolithic rule set. At the macro-level, hand-crafted features decisively outperform neural sequence learning—notably by +0.445 F_1 in 3-way classification—confirming that high-level strategies rely on systematic abstractions that Bi-LSTMs struggle to induce from low-resource data. At the micro-level, M+P significantly outperforms the Bi-LSTM in 4/7 mutation-specific tasks, while three domains (Medial A, Ablaut, Templatic) show no significant difference ($p < 0.05$), suggesting these patterns are equally accessible to either approach. High complementarity further indicates that feature-based and memory-based models capture distinct error patterns, supporting hybrid architectures for non-concatenative morphological analysis.

6.2. The Grammar-Memory Interface

Our findings are consistent with gradient productivity accounts in which computation and storage occupy a continuum rather than categorical routes (Bybee and McClelland, 2005; Hay and Baayen, 2005; O'Donnell, 2015). M+P dominates 7/10 tasks, indicating Tashlhiyt plural formation is more systematic than traditionally assumed. High complementarity in domains like Ablaut (0.520) reveals non-redundant signals: hand-crafted features isolate prosodic structure, while the Bi-LSTM captures patterns where explicit generalizations fail. In this low-resource setting, the Bi-LSTM cannot induce the abstract regularities that features encode directly, explaining its macro-level failure; its micro-level advantages reflect sensitivity to less productive, item-level patterns. High irreducible error (27.7%) alongside low idiosyncrasy severity (<5%) suggests complexity stems from competition among overlapping regularities, consistent with the low conditional entropy conjecture (Ackerman and Malouf, 2013; Cotterell et al., 2019), rather than arbitrary lexical listing.

6.3. Implications for Morphological NLP

Hand-crafted features decisively outperform neural baselines in macro-level tasks (>40pp F_1), demonstrating that sequence models trained from scratch on limited data struggle to induce abstract Amazigh morphological regularities without explicit structural bias. This confirms that explicit linguistic resources are a technical necessity for under-resourced Afroasiatic varieties (Khalifa et al., 2020). Ablation results also suggest that data augmentation for Tashlhiyt must respect prosodic structure, unlike unconstrained generation common in low-resource inflection (Anastasopoulos and Neubig, 2019). For low-resource non-concatenative systems, linguistically informed features are as critical as model architecture.

These limitations on neural models, however, need not be permanent. The annotation schema developed here (e.g., plural class labels, prosodic weight features, foot structure, derivational categories) can serve as a reusable template for related varieties like Central Atlas Tamazight and Tarifit, which share non-concatenative processes but differ in pattern inventories. The systematic singular–plural correspondences could also bootstrap pre-annotation tools that suggest candidate classes for new items, easing community-driven documentation. As structured datasets grow, neural models may acquire sufficient signal to learn regularities they currently fail to induce from raw forms (Kann and Schütze, 2016), making hand-crafted features a bridge to, rather than a substitute for, data-driven approaches.

7. Limitations

Dataset Scope and Speaker Bias. While our dataset of 1,185 nouns is substantial for a low-resource language, its composition introduces potential biases. A significant portion of the paradigms is derived from Jebbour (1996), which reflects the internal grammar of a single native speaker. While these data are supplemented by corpus resources, they may not fully capture the range of cross-dialectal and intra-speaker variation known to exist in Amazigh plural marking. Consequently, our findings on lexical idiosyncrasy may partially reflect speaker-specific patterns rather than language-wide irregulars.

Depth of Phonotactic Modeling. Our feature engineering focuses on prosodic weight, syllable structure, and edge-aligned n-grams. However, we do not explicitly model more abstract phonological interactions, such as vowel-to-vowel co-occurrence constraints or CV-templates (e.g., the root-and-pattern logic dominant in Semitic). Prior work on Arabic suggests that these templatic configurations are important to plural learnability (Dawdy-Hesterberg and Pierrehumbert, 2014). The absence of these features may explain some of the residual variance in our micro-level models, particularly in the prediction of vowel ablaut.

Refining Residual and Error Analysis. Our quantification of residual lexical idiosyncrasy currently defines idiosyncrasy as the intersection of failure across distributional and featural models. A more comprehensive approach would incorporate feature-space minimal pair analysis, identifying items with near-identical feature vectors but divergent class labels; the density of such local ambiguities provides a more precise diagnostic of irreducible idiosyncrasy than global accuracy alone. Furthermore, a qualitative error analysis is needed to determine if misclassifications cluster around specific phonological environments (e.g., specific vowel combinations or consonant clusters) not currently captured in our feature set. Identifying such clusters would distinguish between true lexical exceptions and systematic patterns overlooked by the current model.

Methodological Constraints. To ensure a fair comparison across feature sets, we employed fixed hyperparameters across all models. While this protocol minimizes optimization artifacts, it likely results in sub-optimal performance metrics. Furthermore, our Bi-LSTM baseline represents what character-level sequence models can achieve when trained from scratch at this data scale. More powerful approaches, including pretrained multilingual models, cross-lingual transfer from Afroasiatic languages, or data augmentation techniques (Anastasopoulos and Neubig, 2019), could narrow the performance gap. Our results therefore establish a lower bound for neural approaches rather than an absolute ceiling on sequence model capacity. However, adopting such methods would shift the experimental question from what can be induced from the data alone to what can be transferred from external resources, confounding the controlled comparison that is the paper's central contribution.

8. Ethics Statement

Linguistic Representation and Data Use. This research focuses on Tashlhiyt, a low-resource Amazigh language. Our primary goal is to improve the computational visibility of underrepresented Afroasiatic languages. The dataset used in this study is derived from published lexicographic sources and native speaker intuitions; it contains no personally identifiable information (PII) or sensitive data. By releasing the annotated dataset and feature sets, we aim to provide a public resource for the Amazigh community and researchers to foster further developments in indigenous language technologies.

Annotator Labor. The manual annotation and adjudication performed in this study were conducted by the second author, who was compensated for his labor consistent with local labor standards and academic research norms. No external crowdsourcing or unpaid labor was used in the production of the feature sets.

AI Assistance. We utilized AI assistants (specifically Anthropic's Claude) to generate, integrate, and orchestrate some Python scripts used for experimental analysis, as well as to provide editorial suggestions for prose clarity. All code logic, data processing pipelines, and textual revisions were manually verified and authorized by the human authors, who bear full responsibility for the methodology and content.

Environmental Impact. The computational experiments described in this study, including 10-fold cross-validation and feature selection across 10 domains, were conducted on consumer-grade hardware (Apple M4 Pro). The total training time was under four hours. Given the efficiency of the models used (Logistic Regression, Random Forest, and Gradient Boosting), the carbon footprint of this research is negligible compared to the training of large-scale language models.

9. Acknowledgements

This article has benefited from comments and suggestions from Karim Bensoukas, Abdelkrim Jebbour, Rachid Ridouane, and three anonymous SLiDE reviewers. It was supported by a Social Science and Humanities Research Council of Canada grant (435-2020-0193).

10. Bibliographical References

Farrell Ackerman and Robert Malouf. 2013. Morphological organization: The low conditional entropy conjecture. *Language*, 89(3):429–464.

Wafia Adouane, Nasredine Semmar, and Richard Johansson. 2016a. Romanized berber and romanized arabic automatic language identification using machine learning. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)*, pages 53–61.

Wafia Adouane, Nasredine Semmar, Richard Johansson, and Victoria Bobicev. 2016b. Automatic detection of arabicized berber and arabic varieties. In *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)*, pages 63–72.

Samir Amri, Lahbib Zenkouar, and Mohamed Outahajala. 2017. Build a morphosyntaxically annotated amazigh corpus. In *Proceedings of the 2nd international Conference on Big Data, Cloud and Applications*, pages 1–7.

Antonios Anastasopoulos and Graham Neubig. 2019. Pushing the limits of low-resource morphological inflection. *arXiv preprint arXiv:1908.05838*.

Rkia Bani, Samir Amri, Lahbib Zenkouar, and Zouhair Guennoun. 2023. Deep neural networks for part-of-speech tagging in under-resourced amazigh. *Revue d'Intelligence Artificielle*, 37(3):611.

André Basset. 1952. *La langue berbère*. Oxford University Press.

Samir Ben Si Saïd. 2014. *De la nature de la variation diatopique en kabyle: Étude de la formation des singulier et pluriel nominaux*. Doctoral dissertation, Université Nice Sophia Antipolis.

Samir Ben Si Saïd. 2020. La morphologie nominale et les segments flottants en kabyle. *Multilinguales*, 13.

Carl Darling Buck. 2008. *A dictionary of selected synonyms in the principal Indo-European languages*. University of Chicago Press.

Joan Bybee and James L McClelland. 2005. Alternatives to the combinatorial paradigm of linguistic theory based on domain general principles of human cognition. *Linguistic Review*, 22.

Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

Ryan Cotterell, Christo Kirov, Mans Hulden, and Jason Eisner. 2019. On the complexity and typology of inflectional morphological systems. *Transactions of the Association for Computational Linguistics*, 7:327–342.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016. The sigmorphon 2016 shared task—morphological reinflection. In *Proceedings of the 14th SIGMORPHON workshop on computational research in phonetics, phonology, and morphology*, pages 10–22.

Lisa Garnand Dawdy-Hesterberg and Janet Breckenridge Pierrehumbert. 2014. Learnability and generalisation of Arabic broken plural nouns. *Language, cognition and neuroscience*, 29(10):1268–1282.

François Dell and Mohamed Elmedlaoui. 2002. *Syllables in Tashlhiyt Berber and in Moroccan Arabic*, volume 2. SKluwer Academic Publishers.

Rebecca Hasselbach. 2007. External plural markers in Semitic: A new assessment. In Cynthia L. Miller, editor, *Studies in Semitic and Afroasiatic Linguistics Presented to Gene B. Gragg*, pages 123–138. The Oriental Institute of the University of Chicago.

Jennifer B Hay and R Harald Baayen. 2005. Shifting paradigms: Gradient structure in morphology. *Trends in cognitive sciences*, 9(7):342–348.

Ali Idrissi. 2000. On Berber plurals. In Jacqueline Lecarme, Jean Lowenstamm, and Ur Shlonsky, editors, *Research in Afroasiatic Grammar*, pages 101–124. John Benjamins.

Abdelkrim Jebbour. 1988. Process of formation of the nominal plural in Tamazight (Tachelhiyt of Tiznit, Morocco): Non-concatenative approach. D.E.S. thesis, Mohamed V University, Faculty of Letters, Rabat.

Abdelkrim Jebbour. 1996. *Morphologie et contraintes prosodiques en berbère (Tachelhit de*

Tiznit) — Analyse linguistique et traitement automatique. Doctorat d'état dissertation, Mohammed V University, Faculty of Letters and Human Sciences, Rabat.

Katharina Kann and Hinrich Schütze. 2016. MED: The LMU system for the SIGMORPHON 2016 shared task on morphological reinflection. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 62–70.

Salam Khalifa, Nasser Zalmout, and Nizar Habash. 2020. Morphological analysis and disambiguation for Gulf Arabic: The interplay between resources and methods. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3895–3904.

Christo Kirov and Ryan Cotterell. 2018. Recurrent neural networks in linguistic theory: Revisiting Pinker and Prince (1988) and the past tense debate. *Transactions of the Association for Computational Linguistics*, 6:651–665.

Mena B Lafkioui. 2008. Dialectometry analyses of berber lexis. *Folia Orientalia*, 44:71–88.

Ahmed Lasri. 1991. *Aspects de la phonologie nonlinéaire du parler berbère chleuh de Tidli*. Doctoral dissertation, Université Paris 3.

John J McCarthy. 1979. *Formal problems in Semitic phonology and morphology*. Routledge.

Kate McCurdy, Sharon Goldwater, and Adam Lopez. 2020. Inflecting when there's no majority: limitations of encoder-decoder neural networks as cognitive models for German plurals. *arXiv preprint arXiv:2005.08826*.

Timothy J O'Donnell. 2015. *Productivity and reuse in language: A theory of linguistic computation and storage*. MIT Press.

Hanae Raiss and Violetta Cavalli-Sforza. 2012. Anmorph: Amazigh nouns morphological analyzer. In *Press in Proceedings of the 5th Int. Conf. on Amazigh and ICT*.

Jilali Saib. 1986. Noun pluralization in Berber: A study in internal reconstruction. *Languages and Literatures*, 5:109–133.

I Taghbalout, F Ataa Allah, and M El Marraki. 2015. Amazigh noun inflection in the universal networking language. *International Journal of Education and Information Technology*, 9:122–128.

Catherine Taine-Cheikh. 2006. Alternances vocaliques et affixations dans la morphologie nominale du berbère : le pluriel en zénaga. In Dymitr Ibriszimow, Rainer Vossen, and Harry Stroomer, editors, *Études berbères III : Le nom, le pronom et autres articles*, pages 253–267. Rüdiger Köppe.

Ekaterina Vylomova, Jennifer White, Elizabeth Salesky, Sabrina J Mielke, Shijie Wu, Edoardo Maria Ponti, Rowan Hall Maudslay, Ran Zmigrod, Josef Valvoda, Svetlana Toldova, et al. 2020. Sigmorphon 2020 shared task 0: Typologically diverse morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 1–39.

11. Language Resource References

Alderete, John and Agarwal, Piyush and Holubowsky, Kaye and Jebbour, Abdelkrim. 2025. *Tashlhiyt Noun Paradigm Database: 1914 Paradigms with 50 Linguistic Attributes*. Database of 1914 noun paradigms with phonological, morphological, and semantic attributes for gender and plural prediction.

Jebbour, Abdelkrim and Boukous, Ahmed and El Alami, Abdelkrim and Li, Jane S. Y. and Ridouane, Rachid and Alderete, John. 2021. *Nine Tashlhiyt Texts: Structured Representations of 18,000 Words (First Release)*. Dataset.

Appendices

A. Features

Table 3: Overview of all target variables and features used in the Tashlhiyt plural prediction experiments. Target variables (y) are divided into macro-level strategies and micro-level mutation patterns. Features (m, s, p) include both intrinsic database properties and engineered prosodic/distributional markers.

Feature Type	Feature Label	Variable Type	Vocab Size
Target (y) (Macro)	`y_macro_suffix`	binary	2
	`y_macro_mutated`	binary	2
	`y_macro_3way`	multi-class	3
Target (y) (Micro)	`y_micro_medial_a`	binary	2
	`y_micro_final_a`	binary	2
	`y_micro_final_vw`	binary	2
	`y_micro_ablaut`	binary	2
	`y_micro_final_t`	binary	2
	`y_micro_templatic`	binary	2
	`y_micro_8way`	multi-class	8
Morphological	`m_gender`	binary	2
	`m_deriv_cat`	categorical	6
	`m_paradigm_struc`	categorical	3
	`m_r_augment`	categorical	3
	`m_loan_status`	binary	2
Semantic	`s_animacy`	binary	2
	`s_humanness`	binary	2
	`s_semantic_field`	categorical	22
Phonological (Prosodic & Distributional)	`p_ngrams_macro`	multi-category	2,019
	`p_LH_ends_L`	binary	2
	`p_LH_initial_weight`	binary	2
	`p_LH_less_2_syllables`	binary	2
	`p_LH_all_heavy`	binary	2
	`p_LH_count_heavies`	binary split	2
	`p_LH_count_moras`	continuous	1–10
	`p_foot_count_feet`	continuous	0–3
	`p_foot_residue_right`	binary	2
	`p_foot_residue`	binary	2

B. Reproducibility

Computing Infrastructure and Software. All experiments were conducted on a single Apple M4 processor (macOS 24.3.0). The total compute time for the full 10-domain suite—including feature selection, ablation studies, and stratified 10-fold cross-validation—was under 4 hours. The implementation utilized Python 3.13.2 with the following library versions: `scikit-learn` 1.3.0, `xgboost` 2.0.3, `imbalanced-learn` 0.14.1, and `numpy` 1.24.3.

Fixed Hyperparameter Specifications. To ensure that performance deltas reflect feature quality rather than optimization bias, we utilized identical hyperparameters across all 10 domains. All models used a fixed seed (`random_state`=42) for reproducibility.

- **Logistic Regression:** Inverse regularization strength $C = 1.0$; penalty: L_2; solver: `lbfgs`; `max_iter`=1000; `class_weight='balanced'`.
- **Random Forest:** $n\_estimators = 100$; `max_depth`=10; `min_samples_split`=2; `max_features='sqrt'`; `class_weight='balanced'`.
- **XGBoost:** $n\_estimators = 100$; `learning_rate`=0.1; `max_depth`=10; `subsample`=1.0; `colsample_bytree`=1.0. The `scale_pos_weight` was calculated per domain as $N_{\text{neg}}/N_{\text{pos}}$.
- **Bi-LSTM (Baseline):** Embedding dimension: 32; Hidden units: 64 (bidirectional); Dropout: 0.3; Optimizer: Adam; Early stopping patience: 10 epochs.

Class Imbalance and Validation. Models were evaluated using stratified 10-fold cross-validation. To address extreme class imbalance (minority class $< 10\%$), we applied the Synthetic Minority Over-sampling Technique (SMOTE) to the training folds of the `final_a`, `final_vw`, and `insert_c` domains. SMOTE was configured with `sampling_strategy='auto'` and an adaptive k-neighbors parameter set to $\min(5, N_{\text{minority}} - 1)$. Crucially, over-sampling was restricted to training folds to prevent data leakage, ensuring validation folds remained representative of the original data distribution.

Dataset and Task Domains. The full feature set and target variable mapping are detailed in Table 3. Macro-level domains include $N = 1,185$ nouns, while micro-level domains are restricted to the $N = 562$ nouns exhibiting internal or mixed plural patterns. All target variables were encoded as integers, and categorical features were one-hot encoded for the Logistic Regression baseline. The anonymized annotated dataset is included as supplementary material with this submission.

Feature Selection and Data Leakage Mitigation We acknowledge that performing n-gram feature selection on the full dataset before cross-validation introduces mild information leakage, as feature selection uses target labels from test folds. However, several factors mitigate this concern: (1) feature selection is performed *once* to define a stable feature space, not iteratively tuned per fold; (2) all model weights are learned exclusively on training data; (3) the selected features form a domain-general master set used across all 10 tasks, not task-specific tuning; and (4) our scientific claims focus on *relative* performance between feature sets (which is unaffected by this design choice) rather than absolute performance estimates. Future work could validate this approach by performing nested feature selection within each training fold.

C. Detailed Model Performance

Tables 4–13 report mean percentage ± standard deviation from stratified 10-fold cross-validation. Model architectures: LogReg (Logistic Regression), RanFor (Random Forest), and XGB (XGBoost). Feature sets: Sem (Semantic), Morph (Morphological), Phon (Phonological), M+P (Morphological and Phonological), All F's (All features). Notation: † denotes SMOTE application; ‡ denotes high cross-fold variation ($\sigma \geq 15.0$). See the Reproducibility section for fixed hyperparameter specifications. Tables 4–6 cover macro-level domains ($N = 1,185$); Tables 7–13 cover micro-level domains ($N = 562$). Table 14 reports significance tests.

Table 4: Performance: **Macro: Has Suffix**

Set	Model	Acc	F_1	AUC
Sem	LogReg	52.2 ± 3.9	60.2 ± 5.0	55.2 ± 4.2
	RanFor	56.1 ± 5.7	63.5 ± 7.2	60.5 ± 5.1
	XGB	58.2 ± 6.3	66.4 ± 6.5	60.4 ± 6.2
Morph	LogReg	55.5 ± 4.0	62.1 ± 4.0	57.1 ± 3.9
	RanFor	62.3 ± 8.1	70.1 ± 8.5	63.3 ± 5.9
	XGB	60.0 ± 6.8	67.5 ± 8.0	64.0 ± 6.4
Phon	LogReg	81.9 ± 4.0	87.1 ± 2.7	87.8 ± 4.9
	RanFor	69.9 ± 4.4	76.4 ± 3.9	79.0 ± 4.2
	XGB	73.4 ± 2.8	79.8 ± 2.2	81.2 ± 4.8
M+P	LogReg	82.8 ± 4.4	**87.6 ± 3.0**	88.9 ± 4.9
	RanFor	73.8 ± 4.5	79.9 ± 3.7	80.8 ± 4.2
	XGB	77.0 ± 3.6	82.9 ± 2.3	84.7 ± 4.5
All F's	LogReg	81.5 ± 3.9	86.7 ± 2.6	88.2 ± 4.8
	RanFor	75.8 ± 4.2	81.8 ± 3.4	81.5 ± 4.1
	XGB	77.4 ± 2.3	83.0 ± 1.6	84.3 ± 4.0
N-grams	LogReg	79.6 ± 3.2	85.2 ± 2.2	84.4 ± 5.4
	RanFor	74.5 ± 3.2	80.8 ± 2.2	80.6 ± 4.5
	XGB	73.9 ± 3.6	79.9 ± 2.8	80.9 ± 4.9

Table 5: Performance: **Macro: Has Mutation**

Set	Model	Acc	F_1	AUC
Sem	LogReg	56.7 ± 3.2	56.2 ± 3.3	60.9 ± 3.1
	RanFor	58.9 ± 3.0	63.5 ± 2.6	62.0 ± 3.0
	XGB	58.1 ± 2.2	59.8 ± 2.5	62.0 ± 2.4
Morph	LogReg	56.9 ± 3.3	59.5 ± 2.5	58.7 ± 5.2
	RanFor	57.0 ± 3.1	58.4 ± 3.8	61.1 ± 5.0
	XGB	56.5 ± 3.8	56.9 ± 5.9	61.0 ± 5.2
Phon	LogReg	77.1 ± 4.1	76.6 ± 3.9	84.5 ± 3.7
	RanFor	71.1 ± 4.8	68.3 ± 6.1	79.5 ± 4.1
	XGB	75.3 ± 3.4	74.7 ± 3.2	83.6 ± 3.7
M+P	LogReg	77.7 ± 3.4	**76.9 ± 3.2**	84.9 ± 3.8
	RanFor	73.8 ± 5.1	71.8 ± 5.3	81.3 ± 3.6
	XGB	76.3 ± 3.9	75.3 ± 3.7	85.2 ± 3.0
All F's	LogReg	77.0 ± 2.1	76.3 ± 1.7	84.5 ± 3.4
	RanFor	74.0 ± 3.7	71.6 ± 4.6	80.8 ± 3.6
	XGB	75.6 ± 2.9	74.6 ± 2.8	84.6 ± 2.8
N-grams	LogReg	75.4 ± 3.6	74.7 ± 3.7	82.9 ± 3.4
	RanFor	71.1 ± 3.4	74.5 ± 2.4	79.5 ± 4.3
	XGB	71.5 ± 4.2	73.7 ± 3.7	80.7 ± 3.4

Table 6: Performance: **Macro: 3-way Class.**

Set	Model	Acc	F_1	AUC
Sem	LogReg	41.9 ± 4.8	40.2 ± 4.4	60.6 ± 4.8
	RanFor	43.7 ± 6.7	42.1 ± 6.6	62.1 ± 6.0
	XGB	52.5 ± 3.6	31.7 ± 3.5	62.8 ± 6.0
Morph	LogReg	39.9 ± 4.5	37.4 ± 4.6	60.0 ± 5.3
	RanFor	45.7 ± 5.3	44.0 ± 4.9	63.6 ± 4.3
	XGB	52.8 ± 4.1	32.2 ± 3.8	63.9 ± 5.0
Phon	LogReg	70.4 ± 5.3	65.8 ± 5.5	83.5 ± 4.9
	RanFor	61.6 ± 5.9	57.4 ± 6.8	78.9 ± 4.5
	XGB	68.2 ± 5.0	62.1 ± 6.0	82.1 ± 5.1
M+P	LogReg	72.3 ± 5.0	**68.3 ± 5.4**	84.8 ± 4.9
	RanFor	64.8 ± 5.8	60.4 ± 6.1	80.4 ± 4.4
	XGB	69.6 ± 5.1	64.7 ± 5.5	85.1 ± 4.4
All F's	LogReg	70.5 ± 5.8	66.6 ± 5.8	84.3 ± 5.5
	RanFor	63.6 ± 5.5	59.5 ± 5.7	79.5 ± 4.8
	XGB	68.9 ± 6.8	63.9 ± 8.1	84.4 ± 5.1
N-grams	LogReg	67.3 ± 5.6	61.4 ± 5.4	81.2 ± 5.0
	RanFor	60.2 ± 6.2	57.1 ± 5.9	76.9 ± 5.4
	XGB	65.6 ± 6.4	55.3 ± 7.8	79.1 ± 4.5

Table 7: Performance: **Micro: Medial A**

Set	Model	Acc	F_1	AUC
Sem	LogReg	60.2 ± 5.2	26.6 ± 7.9	56.8 ± 11.5
	RanFor	54.1 ± 5.4	30.8 ± 6.6	57.3 ± 9.3
	XGB	30.8 ± 5.7	28.1 ± 3.6	54.6 ± 8.1
Morph	LogReg	55.2 ± 8.4	35.4 ± 8.1	64.8 ± 9.5
	RanFor	55.4 ± 7.9	36.3 ± 7.7	63.4 ± 9.7
	XGB	55.5 ± 8.3	37.7 ± 8.8	64.6 ± 10.6
Phon	LogReg	73.1 ± 4.2	52.7 ± 6.2	84.3 ± 5.2
	RanFor	72.1 ± 5.7	50.7 ± 7.4	81.1 ± 7.5
	XGB	66.0 ± 12.6	48.2 ± 10.2	81.7 ± 6.6
M+P	LogReg	81.3 ± 5.5	**54.9 ± 12.6**	85.1 ± 4.3
	RanFor	78.3 ± 3.8	51.6 ± 10.0	83.9 ± 5.5
	XGB	77.9 ± 4.3	53.7 ± 8.5	84.2 ± 6.2
All F's	LogReg	81.8 ± 3.5	48.7 ± 8.2	83.1 ± 5.9
	RanFor	79.2 ± 3.6	49.9 ± 9.4	82.0 ± 6.6
	XGB	77.2 ± 6.8	51.3 ± 10.3	81.6 ± 8.6
N-grams	LogReg	78.8 ± 4.8	49.0 ± 10.2	82.7 ± 7.8
	RanFor	71.0 ± 5.1	44.2 ± 6.1	78.1 ± 6.4
	XGB	66.0 ± 6.8	44.9 ± 8.9	78.6 ± 8.9

Table 8: Performance: **Micro: Final A** (†)

Set	Model	Acc	F_1	AUC
Sem	LogReg	58.7 ± 7.0	21.4 ± 8.4	55.3 ± 9.5
	RanFor	54.6 ± 8.5	22.4 ± 4.8	55.7 ± 9.9
	XGB	35.1 ± 5.9	23.6 ± 3.3	57.1 ± 9.4
Morph	LogReg	53.0 ± 5.7	31.0 ± 3.9	66.3 ± 5.8
	RanFor	52.5 ± 5.3	29.6 ± 3.7	65.7 ± 6.6
	XGB	52.3 ± 5.2	29.9 ± 3.5	65.5 ± 6.1
Phon	LogReg	52.3 ± 5.4	27.1 ± 5.0	68.1 ± 7.7
	RanFor	49.8 ± 7.1	27.7 ± 5.1	68.4 ± 6.6
	XGB	29.4 ± 2.8	26.1 ± 1.5	71.3 ± 7.8
M+P	LogReg	76.0 ± 4.2	**33.3 ± 11.5**	75.7 ± 7.9
	RanFor	66.2 ± 4.9	34.6 ± 2.5	76.1 ± 5.9
	XGB	63.9 ± 4.7	34.0 ± 3.1	75.6 ± 7.4
All F's	LogReg	84.3 ± 4.2	32.3 ± 16.5‡	74.6 ± 9.2
	RanFor	70.8 ± 6.2	32.5 ± 5.1	73.5 ± 7.2
	XGB	69.9 ± 4.4	30.8 ± 8.0	71.5 ± 8.1
N-grams	LogReg	81.5 ± 4.6	43.1 ± 10.3	74.6 ± 8.2
	RanFor	69.0 ± 8.6	32.1 ± 12.0	69.7 ± 10.9
	XGB	63.3 ± 5.7	32.9 ± 7.6	69.6 ± 8.8

Table 9: Performance: **Micro: Final Vw** (†)

Set	Model	Acc	F_1	AUC
Sem	LogReg	76.3 ± 4.9	18.3 ± 11.9	72.2 ± 19.3‡
	RanFor	73.0 ± 4.9	19.1 ± 7.7	73.9 ± 18.1‡
	XGB	54.6 ± 4.2	12.6 ± 5.2	67.5 ± 18.1‡
Morph	LogReg	79.5 ± 7.2	21.9 ± 13.6	61.9 ± 22.4‡
	RanFor	83.3 ± 7.0	26.1 ± 15.6‡	73.3 ± 19.1‡
	XGB	80.3 ± 9.6	23.6 ± 15.3‡	71.3 ± 19.7‡
Phon	LogReg	91.6 ± 2.5	40.1 ± 16.8‡	79.5 ± 17.3‡
	RanFor	73.7 ± 9.2	17.8 ± 9.3	72.7 ± 18.1‡
	XGB	39.3 ± 11.0	11.3 ± 4.1	72.1 ± 16.8‡
M+P	LogReg	92.3 ± 2.4	**33.2 ± 20.5**‡	82.5 ± 17.7‡
	RanFor	91.6 ± 3.7	44.1 ± 25.1‡	79.3 ± 18.3‡
	XGB	88.8 ± 5.3	35.0 ± 17.7‡	72.8 ± 20.1‡
All F's	LogReg	93.2 ± 2.6	37.9 ± 24.1‡	82.8 ± 16.1‡
	RanFor	90.7 ± 4.3	40.3 ± 22.6‡	78.1 ± 18.6‡
N-grams	LogReg	89.7 ± 3.0	27.3 ± 21.5‡	81.3 ± 12.7
	RanFor	82.4 ± 3.9	23.4 ± 15.4‡	81.6 ± 13.3
	XGB	68.9 ± 6.8	18.3 ± 7.6	80.9 ± 11.3

Table 10: Performance: **Micro: Ablaut**

Set	Model	Acc	F_1	AUC
Sem	LogReg	58.4 ± 5.0	59.2 ± 4.4	64.3 ± 7.3
	RanFor	60.3 ± 6.0	58.7 ± 5.6	63.8 ± 7.4
	XGB	61.0 ± 5.8	60.3 ± 5.4	64.4 ± 6.8
Morph	LogReg	66.4 ± 6.9	62.9 ± 8.1	70.7 ± 6.8
	RanFor	64.6 ± 6.1	62.5 ± 7.4	70.8 ± 7.1
	XGB	63.9 ± 5.3	62.3 ± 6.9	70.9 ± 6.9
Phon	LogReg	77.6 ± 4.4	76.5 ± 4.6	83.1 ± 5.5
	RanFor	71.9 ± 5.0	70.9 ± 5.1	80.7 ± 6.2
	XGB	75.2 ± 5.7	74.4 ± 5.9	81.9 ± 7.0
M+P	LogReg	76.1 ± 6.3	**75.6 ± 5.7**	83.9 ± 5.9
	RanFor	72.2 ± 5.4	72.5 ± 5.4	81.9 ± 6.3
	XGB	78.6 ± 8.0	77.6 ± 8.7	85.8 ± 5.7
All F's	LogReg	74.9 ± 7.0	73.8 ± 6.9	83.2 ± 6.7
	RanFor	71.9 ± 5.8	72.4 ± 4.8	81.0 ± 6.6
	XGB	77.2 ± 8.5	76.2 ± 8.7	84.7 ± 6.6
N-grams	LogReg	74.0 ± 4.9	73.5 ± 4.4	82.0 ± 5.0
	RanFor	72.4 ± 4.7	71.3 ± 4.7	80.3 ± 5.6
	XGB	74.0 ± 4.9	72.9 ± 5.1	80.8 ± 5.7

Table 11: Performance: **Micro: Final T** (†)

Set	Model	Acc	F_1	AUC
Sem	LogReg	74.4 ± 5.0	11.1 ± 10.3	55.2 ± 15.7‡
	RanFor	61.2 ± 6.8	12.3 ± 7.0	53.6 ± 14.5
	XGB	36.3 ± 9.6	13.2 ± 6.1	56.3 ± 17.2‡
Morph	LogReg	69.9 ± 6.3	21.8 ± 4.7	66.1 ± 8.3
	RanFor	75.1 ± 5.6	26.0 ± 8.0	69.0 ± 11.2
	XGB	75.5 ± 5.6	28.6 ± 7.6	71.0 ± 13.9
Phon	LogReg	73.3 ± 6.5	30.7 ± 8.0	86.9 ± 7.1
	RanFor	64.4 ± 10.3	25.2 ± 7.0	86.9 ± 8.0
	XGB	56.1 ± 6.3	22.2 ± 4.5	87.0 ± 7.5
M+P	LogReg	85.9 ± 4.0	**35.0 ± 17.8**‡	88.0 ± 6.1
	RanFor	80.1 ± 5.9	32.4 ± 7.3	87.2 ± 4.8
	XGB	81.1 ± 4.1	31.2 ± 8.4	85.5 ± 8.1
All F's	LogReg	89.5 ± 3.2	35.3 ± 13.7	84.7 ± 6.7
	RanFor	84.7 ± 4.9	37.3 ± 12.2	87.0 ± 7.0
	XGB	86.5 ± 5.3	40.4 ± 14.9	87.7 ± 6.9
N-grams	LogReg	87.0 ± 3.9	39.9 ± 13.4	89.5 ± 6.8
	RanFor	79.2 ± 5.7	29.8 ± 17.2‡	88.1 ± 7.1
	XGB	71.5 ± 5.3	29.2 ± 8.0	86.0 ± 7.9

Table 12: Performance: **Micro: Templatic**

Set	Model	Acc	F_1	AUC
Sem	LogReg	68.7 ± 6.5	42.5 ± 10.9	77.1 ± 7.8
	RanFor	70.8 ± 6.6	44.0 ± 12.1	75.5 ± 8.1
	XGB	72.1 ± 7.1	44.3 ± 12.0	76.0 ± 7.6
Morph	LogReg	83.6 ± 4.5	68.7 ± 6.4	92.7 ± 3.5
	RanFor	84.2 ± 4.3	69.2 ± 6.4	92.2 ± 4.1
	XGB	85.1 ± 4.4	70.4 ± 6.4	92.8 ± 2.6
Phon	LogReg	93.2 ± 2.8	81.9 ± 7.5	94.1 ± 4.4
	RanFor	91.1 ± 3.3	75.3 ± 10.8	91.9 ± 4.5
	XGB	91.6 ± 2.9	77.1 ± 9.4	93.0 ± 4.4
M+P	LogReg	94.1 ± 1.9	**85.0 ± 4.6**	97.9 ± 1.5
	RanFor	92.7 ± 3.3	83.3 ± 6.8	97.6 ± 1.7
	XGB	95.9 ± 2.1	89.5 ± 5.3	97.5 ± 1.9
All F's	LogReg	94.3 ± 2.0	85.5 ± 4.9	97.8 ± 1.3
	RanFor	92.2 ± 3.6	82.1 ± 7.6	97.7 ± 1.6
	XGB	94.0 ± 2.4	84.4 ± 5.8	97.2 ± 1.6
N-grams	LogReg	92.2 ± 3.6	79.3 ± 9.3	93.9 ± 6.0
	RanFor	92.2 ± 3.3	78.3 ± 8.7	92.1 ± 5.9
	XGB	92.4 ± 2.5	79.7 ± 6.2	92.0 ± 6.7

Table 13: Performance: **Micro: 8-way Class.**

Set	Model	Acc	F_1	AUC
Sem	LogReg	22.2 ± 4.6	17.2 ± 6.0	65.5 ± 6.1
	RanFor	22.2 ± 5.1	17.2 ± 5.6	67.0 ± 4.9
	XGB	45.4 ± 2.3	14.2 ± 2.1	66.8 ± 6.7
Morph	LogReg	31.3 ± 8.7	19.2 ± 5.2	71.1 ± 6.2
	RanFor	28.6 ± 3.1	22.3 ± 2.7	73.4 ± 4.5
	XGB	50.4 ± 4.5	19.9 ± 4.8	73.9 ± 1.8
Phon	LogReg	60.1 ± 6.3	43.9 ± 6.4	81.7 ± 4.2
	RanFor	48.6 ± 5.5	37.0 ± 4.9	78.1 ± 4.8
	XGB	62.5 ± 6.3	39.8 ± 9.9	82.9 ± 2.7
M+P	LogReg	64.8 ± 9.0	**46.9 ± 8.9**	85.1 ± 6.3
	RanFor	56.4 ± 7.1	42.2 ± 6.2	82.2 ± 5.0
	XGB	67.6 ± 5.4	45.8 ± 10.4	87.2 ± 2.6
All F's	LogReg	64.4 ± 6.9	46.4 ± 7.5	86.4 ± 6.5
	RanFor	56.4 ± 8.0	44.7 ± 10.1	86.0 ± 4.9
	XGB	66.7 ± 7.3	44.9 ± 12.5	88.1 ± 3.6
N-grams	LogReg	57.7 ± 7.0	41.8 ± 6.7	80.8 ± 4.1
	RanFor	45.0 ± 6.3	35.0 ± 7.2	77.6 ± 5.9
	XGB	58.9 ± 6.5	38.6 ± 8.9	81.5 ± 3.7

Domain	M+P vs. N-gr		M+P vs. LSTM	
	Δ	p	Δ	p
Macro-Level				
Has Suffix	+0.036*	0.018	+0.062***	<0.001
Has Mutation	+0.023*	0.044	+0.378***	<0.001
3-way	+0.068***	<0.001	+0.445***	<0.001
Micro-Level				
Medial A	+0.037	0.067	+0.042	0.426
Final A	−0.067*	0.048	+0.135***	<0.001
Final V/W	+0.037	0.156	+0.126**	0.005
Ablaut	+0.021	0.197	+0.025	0.180
Insert C	−0.027	0.547	+0.124*	0.013
Templatic	+0.034	0.054	+0.059	0.486
8-way	+0.051**	0.006	+0.383***	<0.001

Table 14: Significance tests (paired two-tailed t-test) comparing M+P against baselines. Δ values denote mean difference in Macro-F_1 across 10 folds. Significance: ***$p < 0.001$, **$p < 0.01$, *$p < 0.05$.

Towards Corpus-Grounded Agentic LLMs for Multilingual Grammatical Analysis

Matej Klemen[1], **Tjaša Arčon**[1], **Luka Terčon**[2], **Marko Robnik-Šikonja**[1], **Kaja Dobrovoljc**[1,2,3]
[1]University of Ljubljana, Faculty of Computer and Information Science, Večna pot 113, 1000 Ljubljana, Slovenia
[2]University of Ljubljana, Faculty of Arts, Aškerčeva cesta 2, 1000 Ljubljana, Slovenia
[3]Jožef Stefan Institute, Jamova cesta 39, 1000 Ljubljana, Slovenia
{matej.klemen, tjasa.arcon, marko.robnik}@fri.uni-lj.si,
{luka.tercon, kaja.dobrovoljc}@ff.uni-lj.si

Abstract

Empirical grammar research has become increasingly data-driven, but the systematic analysis of annotated corpora still requires substantial methodological and technical effort. We explore how agentic large language models (LLMs) can streamline this process by reasoning over annotated corpora and producing interpretable, data-grounded answers to linguistic questions. We introduce an agentic framework for corpus-grounded grammatical analysis that integrates concepts such as natural-language task interpretation, code generation, and data-driven reasoning. As a proof of concept, we apply it to Universal Dependencies (UD) corpora, testing it on multilingual grammatical tasks inspired by the World Atlas of Language Structures (WALS). The evaluation spans 13 word-order features and over 170 languages, assessing system performance across three complementary dimensions – dominant-order accuracy, order-coverage completeness, and distributional fidelity – which reflect how well the system generalizes, identifies, and quantifies word-order variations. The results demonstrate the feasibility of combining LLM reasoning with structured linguistic data, offering a first step toward interpretable, scalable automation of corpus-based grammatical inquiry.

Keywords: corpus linguistics, grammatical analysis, large language models, agentic systems, Universal Dependencies, word-order variation

1. Introduction

In the past four decades, linguistics has undergone a fundamental transformation from intuition-based inquiry to data-driven analysis, enabled by large-scale annotated corpora and computational tools that allow grammatical patterns to be studied systematically across languages and communicative contexts (Leech, 1992; McEnery and Hardie, 2011; Sinclair, 1991). Yet, despite these advances, corpus-based linguistic analysis remains labor-intensive and hypothesis-driven, requiring complex scripting, specialized tools, readily accessible corpora, and methodological decisions that constrain exploratory research (Gries, 2009; Biber, 2009) – limiting the scope of feasible investigations (Kaunisto and Schilk, 2024).

Recent advances in large language models (LLMs) with advanced reasoning capabilities (Srivastava et al., 2023; Wei et al., 2022) open new avenues for addressing these limitations, as they have been shown to capture a broad range of grammatical regularities, perform complex linguistic reasoning, and generate structured, interpretable outputs (Beguš et al., 2025; Jumelet et al., 2025; Xia et al., 2024). Their ability to understand natural-language instructions and interface with external data sources (Lewis et al., 2020) suggests that they can now serve as intelligent mediators between linguistic questions and empirical evidence, bringing a new level of automation and accessibility to corpus-based grammatical analysis.

To illustrate the concept, we use the problem of determining the typical order of subject, verb, and object in a language throughout the paper. In Figure 1, we show the contrast between the steps a user might take when trying to tackle the problem traditionally versus using an LLM-based grammatical analysis agent. In the traditional approach, the user needs to plan the experiment, select a relevant resource (e.g., a parsed corpus), select an appropriate tool (e.g., Python) and learn to use it, then analyze the resource. When using a grammatical analysis agent, the steps are abstracted away from the user, but may be inspected to validate the correctness of the process.

To explore this potential, we describe the evaluation framework to mimic the motivational use-case, and propose a grammatical agent prototype to test the feasibility of the idea. Specifically, we apply it to word-order analysis in the Universal Dependencies (UD) corpora, which provide cross-linguistically consistent grammatical annotation and enable large-scale, data-driven studies of grammatical structure and variation (de Marneffe et al., 2021; Nivre et al., 2020). More concretely, our contributions are:

1. **An exploratory evaluation framework for corpus-grounded grammar analysis.** We

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 136–147

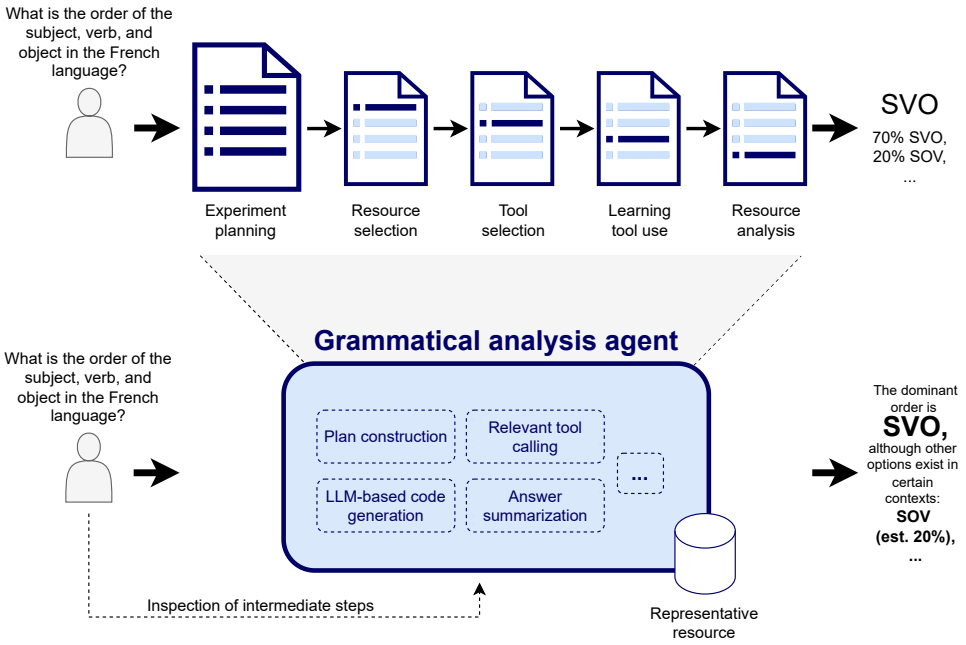


Figure 1: A simplified representation of the contrast between a traditional approach to corpus-based linguistics research (top), and our abstract solution proposal (Grammatical analysis agent, bottom).

define a structured setup for evaluating grammatical analysis agents on word-order phenomena in UD data. The current demonstration framework covers 13 tasks across up to 179 languages, each with defined expected outputs. To enable automatic evaluation, we use classification accuracy, F_1 score, and Hellinger distance to measure different aspects of distributional grammatical analysis (dominant pattern prediction, coverage of relevant patterns, and pattern distribution prediction).

2. **Baseline evaluation.** Using our framework, we test a strong baseline model (GPT-5) that approaches the tasks without explicit resource-grounding, instead relying on its pre-existing linguistic knowledge. We show that the model contains a varying amount of linguistic knowledge depending on the problem, but in general leaves significant room for improvement.

3. **Grammatical analysis agent prototype.** To verify the potential of resource grounding, we develop a prototype grammatical analysis agent (UDagent) with a simple agentic architecture: a fixed plan, a GPT-5-based code generation task-solving module, and answer construction based on the UD data. We show that despite being a conceptually simple prototype, the system already shows a performance improvement on most tasks.

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 introduces the task and evaluation framework including the design of the evaluation dataset. Section 4 describes the architecture of our agentic LLM system. Section 5 presents evaluation results across models, features, and languages. Section 6 concludes with future directions.

2. Related Work

Corpus-based approaches have transformed grammatical research by enabling systematic analyses of authentic language data to uncover patterns, variation, and change (Biber et al., 2021). The growing availability of grammatically annotated corpora, such as those developed within the Universal Dependencies framework (de Marneffe et al., 2021), has been instrumental in advancing research across linguistic typology (Levshina, 2019; Yan and Liu, 2023; Gerdes et al., 2021), data-driven grammar induction (Herrera et al., 2024), cross-linguistic comparison of structural patterns (Futrell et al., 2015), and the study of syntactic complexity (Berdicevskis et al., 2018). These developments have established a range of computational methods that require specialized technical and statistical expertise to extract and analyze grammatical patterns from parsed corpora.

More recently, advances in large language models have opened new possibilities for interacting with linguistic corpora through natural language.

Initial applications demonstrate this potential in different contexts: the integration of LLMs into English-Corpora.org (Davies, 2025) enables users to send corpus outputs such as frequency lists or concordance lines to an LLM for summarization, explanation, and interpretation; CorpusChat supports interactive language learning through corpus-informed dialogue grounded in authentic usage examples (Cheung and Crosthwaite, 2025), highlighting the complementarity between corpus-based and generative AI approaches (Crosthwaite and Baisa, 2023); and text-to-CQL systems translate natural-language requests into formal corpus queries (Lu et al., 2024). Beyond querying, LLMs have also shown promise in explaining linguistic patterns (Singh et al., 2023) and inducing grammatical rules from corpus data (Spencer and Kongborrirak, 2025). However, most of these applications remain task-specific and limited in scope, lacking a general, systematically evaluated framework that combines natural-language prompting with corpus-grounded reasoning to support large-scale grammatical analysis.

More broadly, the recent rise of agentic LLMs has transformed how language models are used for research, enabling them to autonomously plan tasks, access external data, and perform complex analyses across disciplines. The ReAct paradigm (Yao et al., 2023) introduced an explicit coupling of reasoning and acting, allowing LLMs to interleave natural language reasoning steps with concrete tool- or environment-driven actions. Further work has focused on refining and expanding the agentic components, often focusing on a particular use case such as data analysis (Guo et al., 2024) or machine learning (Liu et al., 2025). In our work, we explore grammatical analysis as a new application for LLM-based agents, present a prototype system UDagent, and an evaluation framework along with a multilingual dataset to support further development.

3. Task and Evaluation Framework

Building on the conceptual framework presented in Figure 1, this section describes how the approach was instantiated and evaluated on word-order phenomena in the Universal Dependencies (UD) corpora. We first define the grammatical task at hand (Section 3.1), describe the evaluation dataset and its cross-linguistic coverage (Section 3.2), and then introduce the evaluation metrics used to assess system performance (Section 3.3).

3.1. Task Definition

The evaluation framework builds on typological distinctions captured in the World Atlas of Language Structures (WALS; Dryer and Haspelmath (2013)) which documents cross-linguistic variation across hundreds of grammatical features based on evidence from descriptive grammars. WALS encodes how individual languages realize specific linguistic properties, assigning each language one of several predefined categorical values. For example, for the feature *Order of Object and Verb*, each language is classified as *VO* (the object follows the verb), *OV* (the object precedes the verb), or *No dominant order*.

We take this typological framework as a starting point but extend it to reflect patterns actually attested in corpus data. While WALS adopts discrete, classification-oriented labels, many languages exhibit within-language variation that is better captured through gradient, corpus-based evidence (Yan and Liu, 2023; Levshina et al., 2023; Baylor et al., 2024). Our formulation therefore complements WALS-like grammatical description by producing richer, multi-layered outputs that reveal not only the dominant order but also the diversity and proportional distribution of patterns observed in real usage.

Each grammatical analysis task corresponds to analysing one such word-order relation by receiving as input a grammatical question and a target language, such as: “What is the order of subject, object, and verb in French?” Given such a question and the corresponding language corpus, the system is expected to produce a structured summary of the observed patterns. The output consists of three layers of analysis, listed from least to most detailed:

- **Dominant order:** the most frequent pattern in the corpus (e.g., SVO);
- **Attested orders:** all observed patterns in the corpus (e.g., SVO, SOV, VSO);
- **Order distribution:** proportional frequencies of each pattern in the corpus (e.g., SVO: 92%, SOV: 6%, VSO: 2%)

This three-layered formulation defines grammatical analysis as a structured, data-driven reasoning task that enables evaluation across increasing levels of descriptive completeness. Although developed to support the word-order analyses presented in the continuation of this paper, in further work, the same reasoning framework will be extended to other tasks that examine the distribution and variation of linguistic patterns in corpus data.

3.2. Evaluation Dataset

Building on the word-order analysis framework introduced in Section 3.1, we constructed a large-scale multilingual dataset comprising 13 word-order-related grammatical questions with three-layer ground-truth annotations derived from the UD

WALS ID	Linguistic Task	#Options	#Languages	#Languages$_{>30}$
81A	Order of subject, object and verb	6	169	92
82A	Order of subject and verb	2	177	147
83A	Order of object and verb	2	178	156
84A	Order of object, oblique, and verb	6	165	101
85A	Order of adposition and noun phrase	2	172	133
86A	Order of genitive and noun	2	107	66
87A	Order of adjective and noun	2	163	123
88A	Order of demonstrative and noun	2	104	74
89A	Order of numeral and noun	2	163	101
90A	Order of relative clause and noun	2	158	104
94A	Order of adverbial subordinator and clause	3	160	93
144A	Order of subject, object, verb, and negative word	25	76	18*
144B	Order of negative word and verb relative to clause boundaries	6	94	53

Table 1: Linguistic questions in the evaluation dataset along with the number of possible answer options (#Options), number of languages for which at least one valid answer was produced using the ground truth scripts (#Languages), and number of those for which at least 30 answers were produced (#Languages$_{>30}$).

corpora. UD is a collection of manually annotated treebanks that provide consistent morphosyntactic annotation across typologically diverse languages (de Marneffe et al., 2021), with each word analyzed for lemma, part of speech, morphological features, and syntactic head–dependent relations following a uniform dependency annotation scheme.

Task selection. To align our work with aforementioned typological investigations, we reviewed all 56 word order features documented in WALS, and selected those meeting three criteria: (1) fully representable using UD annotation, (2) not dependent on language-specific lexical information, (3) compatible with our three-step analytical framework. This yielded a total of 13 features, each reformulated as a natural language question for evaluation (e.g., *What is the order of subject, object, and verb in language X?*). Possible answer categories were based on WALS inventories, with minor adaptations to ensure compatibility with UD data and to maintain a consistent analytical framework. For example, due to the complexity of defining the answer "No dominant order", it was currently omitted in favor of a simplified approach that always selects the most frequent order, as described below.

Ground-truth generation. Gold-standard answers were derived automatically from test sets of 179 UD (v2.16; Zeman et al. (2025)) treebanks (one per language) using a human-written Python script. When selecting a treebank for a language, we prioritize larger treebanks that span diverse domains. For each task, the scripts counted all configurations corresponding to a given question and produced two complementary outputs: (1) a list of attested order variations with their frequencies, and (2) a final answer, determined by selecting the most frequent order variation.

Dataset composition. The final evaluation dataset consists of the 13 linguistic questions, their predefined value sets, and the corresponding three-layered gold answers across up to 179 languages, resulting in 1,899 individual feature-language pairs for benchmarking future grammatical analysis systems. The dataset is openly released to support reproducible experimentation and cross-linguistic comparison (Terčon et al., 2026). [1]

Evaluation scope. Table 1 summarizes all linguistic tasks (questions) included in the evaluation dataset, showing their corresponding WALS feature IDs, the number of answer options, and language coverage. The latter includes the number of languages with at least one valid answer – i.e., at least one relevant word-order pattern observed in the corpus. While we release the evaluation data for all languages, we perform the evaluation in our paper only on the languages with reasonable statistical support. We determine the threshold of 30 valid answers as a reasonable trade-off between evaluation on a high number of languages and evaluation with sufficient data support. For feature 144A, the threshold was lowered to 10, reflecting its generally lower frequency across treebanks (marked with an asterisk in Table 1).

[1] We release other supporting resources at https://github.com/matejklemen/ud_llm (modeling code, generated code, and used prompts).

3.3. Evaluation Metrics

During the evaluation phase, the final answers provided by the manually-written Python scripts are treated as the gold standard against which the answers returned by the grammatical analysis system are compared. We adopt a three-dimensional evaluation procedure covering different aspects of system performance: accuracy of the dominant answer, answer coverage completeness, and distributional fidelity.

- To measure the **accuracy of the dominant order** prediction, we use classification accuracy, defined as the ratio between the number of languages with dominant order prediction matching the ground truth, and the total number of evaluated languages.
- To measure the **order coverage completeness**, we use the F_1 score averaged across the evaluated languages. In the expressions below, given the predicted distribution and the correct distribution, the options O_x are the outcomes with relative frequency above zero. Precision, recall, and the coverage F_1 are defined as follows, where L is the set of evaluated languages.

$$Coverage\ F_1 = \frac{1}{|L|} \sum_{L_i \in L} \frac{2 \cdot P_{L_i} \cdot R_{L_i}}{P_{L_i} + R_{L_i}} \quad (1)$$

$$P_{L_i} = \frac{|O_{predicted} \cap O_{correct}|}{|O_{predicted}|} \quad (2)$$

$$R_{L_i} = \frac{|O_{predicted} \cap O_{correct}|}{|O_{correct}|} \quad (3)$$

- To measure the **fidelity of the predicted order distribution**, we use the Hellinger distance (Hellinger, 1909). The distance is bound between 0 and 1, with a lower score indicating better fidelity, i.e. the returned distribution being closer to the manually computed one. The Hellinger distance between two distributions is defined with the following equation, where P is the predicted option distribution and C is the correct option distribution.

$$H(P, C) = \frac{1}{\sqrt{2}} \sqrt{\sum_i (\sqrt{P_i} - \sqrt{C_i})^2} \quad (4)$$

4. Agentic LLM System

In this section, we describe our grammatical analysis agent prototype. An agent may include various components (e.g., planning, code generation, tool calling), helping it adapt to different needs when dealing with different tasks. To test the feasibility of a grammatical analysis agent, we propose a simple architecture to constrain the initial system complexity. We show its conceptual design in Figure 2, and describe its key components next.

Planning. The proposed system uses a fixed plan, converting the initial general task into an example-level subtask, generating code to solve it, applying it to the corpus, and summarizing the results into a concise answer (majority answer) and an extended answer (answer distribution).

Task conversion. To convert the general task into an example-level task, its description is edited to include that the task is to be done on a single sentence. For example, based on the initial task of determining the subject, verb, and object order in French, an example-level subtask of determining the subject, verb, and object order *in the given sentence* is created.

Linguistic processing. Based on the example-level task description, the task solution code is generated using an LLM. Any strong model can be used; in our experiments we resort to GPT-5 (OpenAI, 2025) with high reasoning effort and low answer verbosity as a strong base model. In the prompt, a description of the data format in UD (CONLL-U) is provided along with the instructions to generate a Python function accepting a single sentence as an argument, and returning one or more of the possible valid answers, or None if the question is not answerable. The prompt does *not* include any concrete examples from the treebanks. The valid answer options are included in the prompt due to the specific formulation of WALS tasks to simplify the evaluation procedure. Using this formulation, the instructions guide the model towards generating a general task-solving code untied to a specific language.

Resource analysis. The generated code is executed example-by-example on the chosen UD treebank. In our experiments, we execute the generated code on the training splits of UD treebanks if they exist, and the test splits otherwise. None of the ground truth outputs are exposed to the system during the process, allowing us to use test sets without the risk of data contamination. After code execution, each example is assigned an answer, or the value None if the task is not solvable for a given example. For example, a sentence cannot be assigned a subject, verb, and object order if it does not contain all three elements.

Result summarization. To produce the response, a distribution of the generated example-level answers is computed. The answers with the value None are discarded; thus, the code generation module also implicitly acts as a retrieval filter. The distribution represents the detailed answer returned by the system, while the dominant order prediction is produced by returning the most frequently occurring item in the distribution. For example, if

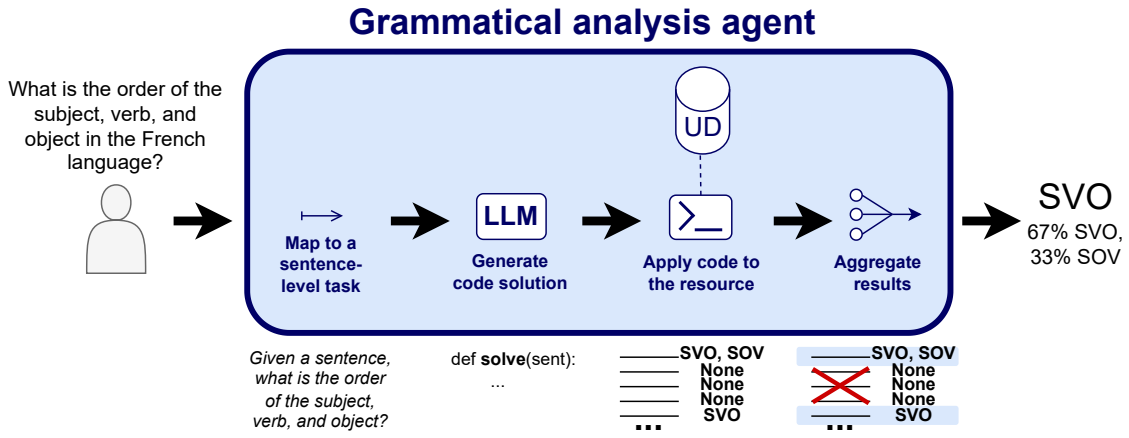


Figure 2: The proposed UDagent system implementation, along with a system trace showing how the system would handle the problem of determining the order of the subject, verb, and object in the French language.

there are 67% examples with the SVO order and 33% with the SOV order, the dominant answer is SVO.

5. Evaluation and Analysis

In this section, we evaluate the proposed grammatical analysis system (UDagent) using the presented evaluation framework. First, we begin with a quantitative analysis across all word-order tasks (Section 5.1), then examine performance patterns across languages (Section 5.2), and finally analyse failure cases to identify limitations of the approach (Section 5.3).

In the quantitative analysis, we measure the accuracy of the computed dominant answer, the coverage completeness, and the distributional fidelity using the metrics presented in Section 3.2. We compare UDagent against two baselines:

- **Majority baseline.** As the evaluated tasks may have an imbalanced ground truth distribution, we include the majority baseline to mitigate the chance for a potential misinterpretation of a high accuracy score. For a given language, the majority baseline predicts the dominant answer for the feature according to WALS. We select the dominant answer from WALS instead of the training distribution as the training set is not available for every language. Its performance can be seen as the lower boundary on acceptable performance.
- **Off-the-shelf GPT-5 baseline.** To test the existing linguistic knowledge in the latest LLMs, and the effect of (not) including external data into a grammatical analysis system, we include the unmodified GPT-5 model (OpenAI, 2025) as a baseline. The model is instructed to jointly return the dominant answer and the distribution of the possible options. We use the model with the reasoning effort set to "high" and the answer verbosity set to "low".

5.1. Overall Performance

We present the automated metric scores for our proposed system and the two baselines in Table 2. Statistical significance of differences in classification accuracy between GPT-5 and UDagent was assessed using the McNemar's test at a significance level of $p < 0.05$ (McNemar, 1947).

Focusing on the baseline scores in isolation, we can observe that the majority baseline scores vary significantly, showing the different degrees of imbalance across tasks. For example, the accuracy of the majority baseline is 0.838 for feature 88A, indicating one order of the demonstrative and noun is dominant for most evaluated languages. The GPT-5 baseline achieves fairly high metric scores across most tasks, achieving the best accuracy score on four, the best coverage F_1 on two, and the best Hellinger distance on three tasks.

In general, we can observe that the proposed system achieves a performance improvement against the baselines in the majority of the tasks. More concretely, it achieves the best accuracy scores in 9, the best coverage F_1 scores in 10, and the best distributional fidelity in 9 out of 13 tasks. In these cases, the achieved accuracy scores and the coverage F_1 scores are very high, in most cases above 0.9. Moreover, the Hellinger distances are low (often below 0.1), indicating the computed distribution closely follows the ground truth distribution. The difference in distribution is caused by differ-

Feature	(Accuracy ↑) Majority	GPT-5	UDagent	(Coverage F_1 ↑) Majority	GPT-5	UDagent	(Hellinger dist. ↓) Majority	GPT-5	UDagent
81A	0.326	0.772	**0.935**	0.301	0.783	**0.826**	0.703	0.242	**0.179**
82A	0.905	0.918	**0.986**	0.687	0.918	**0.968**	0.303	0.147	**0.075**
83A	0.417	0.865	**0.955**	0.660	0.870	**0.955**	0.560	0.151	**0.097**
84A	0.525	0.703	**0.822**	0.302	0.734	**0.908**	0.644	0.349	**0.169**
85A	0.346	0.932	**0.985**	0.531	0.850	**0.924**	0.657	0.105	**0.043**
86A	0.530	**0.970**	0.955	0.636	0.864	**0.904**	0.466	0.101	**0.061**
87A	0.333	0.911	**0.976**	0.572	0.862	**0.962**	0.686	0.136	**0.039**
88A	0.838	0.878	**1.000**	0.815	0.874	**0.937**	0.176	0.131	**0.020**
89A	0.069	**0.931**	**0.931**	0.457	0.756	**0.847**	0.836	0.189	**0.125**
90A	0.721	**0.904**	0.798	0.744	0.795	**0.904**	0.296	**0.150**	0.182
94A	0.871	0.903	**0.936**	0.634	0.664	**0.948**	0.252	0.224	**0.088**
144A	0.278	**0.667**	0.056	0.180	**0.559**	0.028	0.801	**0.419**	0.971
144B	**0.774**	0.642	0.000	0.335	**0.569**	0.000	0.502	**0.408**	1.000

Table 2: Dominant order prediction accuracy, order coverage completeness (F_1), and distributional fidelity (Hellinger distance) scores achieved by the proposed system (UDagent) and the two baselines. The best scores for each task are shown in bold. Statistically significant differences are underlined.

ences between the generated and the ground truth task-solving code. This results in certain examples being mislabeled or incorrectly discarded by the system. In two out of 13 tasks our system fails, and achieves performance worse than the GPT-5 baseline as well as the majority baseline. This can be attributed to the failure of the code generation module which produced code that was syntactically correct but was completely unsuccessful at solving the example-level task (see Section 5.3 for details).

5.2. Cross-Linguistic Analysis

To get a better insight into the performance of UDagent, we analyze its behavior by language, more specifically, its accuracy and coverage. As the system fails universally on features 144A and 144B, we exclude them from analysis, focusing on nontrivial performance instead. To estimate the accuracy for a language, we count the number of correctly solved tasks out of tasks for which a language has the ground truth label assigned.

We observe that our system achieves perfect dominant order accuracy for 112 out of 159 languages; 37 achieve perfect coverage of attested orders in addition to perfect accuracy. The former set involves a diverse selection of languages, hinting towards the strong language independence of the produced code. The latter set mostly involves languages with fewer annotated features, making perfect performance easier to achieve. A notable exception is Czech, with perfect accuracy and coverage for all 11 analyzed features.

At the other end of the spectrum, we observe that the system does not perform terribly even on the worst-ranked languages. For two of the five worst-ranked languages with respect to dominant order prediction accuracy (Sinhala, Yupik), very few tasks with ground truth labels are available (2 and 3, respectively). For the remaining three, the system achieves accuracy of 0.556 and greater:

- In the case of **Buryat** and **Komi Zyrian**, the incorrect answers can be attributed to the very limited size of the UD training splits for these treebanks, which contain few or no instances of the relevant grammatical phenomena.[2]
- In case of **Dutch**, the three incorrect dominant order predictions out of nine stem from imprecise distribution estimates. For features 83A and 84A, the dominant options occur with nearly equal frequency, making the outcome highly sensitive to small distributional shifts. In both cases, UDagent slightly reversed their frequency order (see Figure 3). For feature 81A, the discrepancy reflects a different linguistic interpretation of the task (see Section 5.3): our WALS-inspired ground truth includes only nominal arguments (i.e., subjects and objects expressed as nouns), whereas the agent also counted other parts of speech, such as pronouns – a difference that is not necessarily erroneous given the simple, WALS-agnostic phrasing of the question given in the prompt.

5.3. Error and Failure Analysis

To better understand the causes of performance variation, we conducted a qualitative inspection of the code generated by the system for the five tasks

[2]The small training set size results from a specific UD policy prioritizing the size of the test set when the treebank size is small.

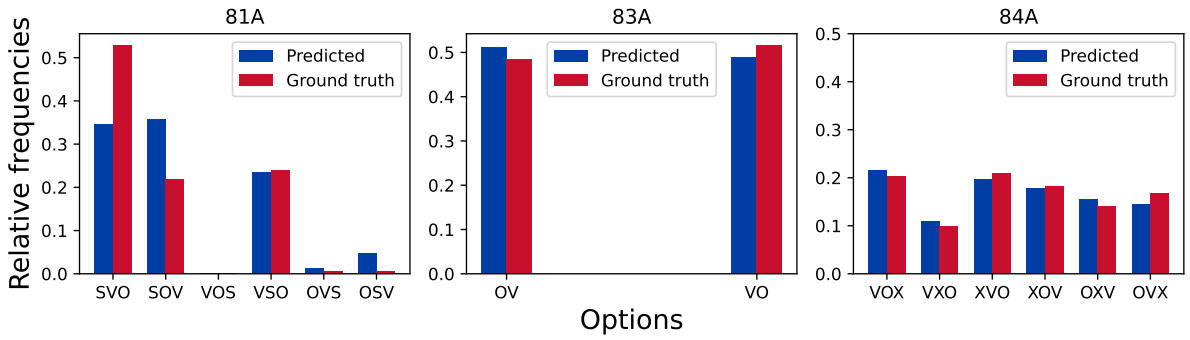


Figure 3: Side-by-side visualization of the predicted and the ground truth intermediate answer distribution for features 81A, 83A, and 84A for Dutch. On features 83A and 84A minor distribution miscalculations result in the wrong dominant answer prediction.

where it performed below the GPT-5 or majority baseline (86A, 89A, 90A, 144A, and 144B). Examining the intermediate scripts provided insight into how the agent interprets linguistic instructions and where the process fails. Two main types of issues emerged: (1) technical errors that prevented the intended operation of the script, and (2) conceptual mismatches in the linguistic interpretation of the task.

The first category includes logic or data-handling errors that did not break Python syntax but produced incomplete or misleading outputs. For instance, in feature 144A (*Order of subject, object, verb, and negative word*), there is an error in the generated code when attempting to read the *Polarity=Neg* UD feature from tokens in the treebank. Consequently, no negative words were identified in any treebank and thus the code could not produce appropriate results. Such local flaws had cascading effects, rendering otherwise sound reasoning ineffective.

In contrast, the second category reflects differences in how the model operationalized grammatical concepts given in the prompt. For example, in feature 86A (*What is the order of genitive and noun?*), both our reference implementation and the agent-generated code identified nominal phrases in which the modifying noun bore the *Case=Gen* feature as potential genitive constructions (among other structures). However, while our ground-truth implementation only considered genitive-like constructions labeled with the *nmod* or *nmod:poss* relations (e.g., *the president's car*), the system also included instances of the *det* relation, which some treebanks use to mark adjectival pronominals expressing possession (e.g. *my car*). As with the Dutch example above, this is not necessarily an error but rather a broader interpretation of the grammatical phenomenon implied by the prompt.

Overall, these analyses highlight two complementary limitations of the current system: the fragility of automatic code generation for complex data operations, and the inherent ambiguity of translating underspecified linguistic definitions into executable form. Both point to the need for future work on incorporating safeguards and clearer linguistic specifications into the prompting and verification stages of grammatical analysis agents. At the same time, the agent's capacity to devise novel computational strategies for solving linguistic problems also reveals an unprecedented opportunity to move beyond hypothesis-driven corpus analysis, opening new avenues for data-driven discovery.

6. Conclusion

This paper presents a framework for corpus-grounded grammatical analysis using agentic LLMs. The proposed system, UDagent, integrates natural-language interpretation, code generation, and corpus-based reasoning, allowing grammatical questions to be answered directly from annotated data. Tested on multilingual word-order variation in UD corpora, the approach proved both scalable and effective, handling over 170 languages with interpretable outputs that closely reflected corpus distributions.

The evaluation showed that even with a relatively simple architecture, the agent achieved strong accuracy and coverage across most tasks, confirming the feasibility of LLM-assisted grammatical research. However, the study has important limitations. First, the evaluation used a single very large proprietary model GPT-5. To test the feasibility of the idea, we limited the evaluation to a state-of-the-art model, while our future work will be focused on improving efficiency and using smaller LLMs. Second, the evaluation covered a handful of elementary linguistic phenomena with uneven language coverage. In future work, we plan to expand the evaluation to more difficult linguistic problems that require complex reasoning. These could be phe-

nomena from alternative databases such as URIEL (Littell et al., 2017) and Grambank (Skirgård et al., 2023), as well as research questions from linguistic scientific papers. Third, the code generation module in our system occasionally failed, producing invalid answers in the process. In future work, we plan to introduce safeguards to reduce the occurrence of such failures. In conclusion, future work should test open and smaller models, expand to multi-step reasoning tasks requiring diverse aggregation strategies, and generalize the architecture. A reasonable expansion is the implementation of external tool calling. For example, instead of attempting to reinvent and outperform specialized word order prediction systems, the agent can be taught to call such tools when needed.

Despite these limitations, the results demonstrate that LLM-based systems can meaningfully bridge linguistic questions and empirical corpus evidence. By making large-scale grammatical analysis more accessible and interpretable, such systems complement traditional linguistic resources and offer new pathways for cross-linguistic discovery, including the enrichment of typological databases such as WALS with usage-based evidence. As agentic architectures and tool-use capabilities continue to advance, the integration of LLM reasoning with structured linguistic data holds significant promise for transforming how we explore and document grammatical knowledge across the world's languages.

Acknowledgments

The work was primarily supported by the Slovene Research and Innovation Agency (ARIS) project GC-0002 and the core research programme P6-0411. The work was also supported by EU through ERA Chair grant no. 101186647 (AI4DH).

7. Bibliographical References

Emi Baylor, Esther Ploeger, and Johannes Bjerva. 2024. Multilingual gradient word-order typology from Universal Dependencies. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 42–49.

Gašper Beguš, Maksymilian Dabkowski, and Ryan Rhodes. 2025. Large linguistic models: Investigating LLMs' metalinguistic abilities. *IEEE Transactions on Artificial Intelligence*, page 1–15.

Aleksandrs Berdicevskis, Çağrı Çöltekin, Katharina Ehret, Kilu von Prince, Daniel Ross, Bill Thompson, Chunxiao Yan, Vera Demberg, Gary Lupyan, Taraka Rama, and Christian Bentz. 2018. Using universal dependencies in cross-linguistic complexity research. *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*.

Douglas Biber. 2009. Corpus-based and corpus-driven analyses of language variation and use. In *The Oxford Handbook of Linguistic Analysis*. Oxford University Press.

Douglas Biber, Stig Johansson, Geoffrey N. Leech, Susan Conrad, and Edward Finegan. 2021. *Grammar of Spoken and Written English*. John Benjamins Publishing Company.

Lisa Cheung and Peter Crosthwaite. 2025. CorpusChat: integrating corpus linguistics and generative AI for academic writing development. *Computer Assisted Language Learning*, 0(0):1–27.

Peter Crosthwaite and Vit Baisa. 2023. Generative AI and the end of corpus-assisted data-driven learning? Not so fast! *Applied Corpus Linguistics*, 3(3):100066.

Mark Davies. 2025. AI/LLM integration with the corpora from English-Corpora.org. Technical report, English-Corpora.org. Last edited 25 August 2025.

Marie-Catherine de Marneffe, Christopher D. Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal dependencies. *Computational Linguistics*, 47(2):255–308.

Richard Futrell, Kyle Mahowald, and Edward Gibson. 2015. Large-scale evidence of dependency length minimization in 37 languages. *Proceedings of the National Academy of Sciences*, 112(33):10336–10341.

Kim Gerdes, Sylvain Kahane, and Xinying Chen. 2021. Typometrics: From Implicational to Quantitative Universals in Word Order Typology. *Glossa: a journal of general linguistics (2021-...)*, 6(1):17.

Stefan Th. Gries. 2009. *Quantitative Corpus Linguistics with R: A Practical Introduction*, 1 edition. Routledge, New York.

Siyuan Guo, Cheng Deng, Ying Wen, Hechang Chen, Yi Chang, and Jun Wang. 2024. DS-agent: Automated data science by empowering large language models with case-based reasoning. In *Proceedings of the 41st International Conference on Machine Learning*.

E. Hellinger. 1909. Neue begründung der theorie quadratischer formen von unendlichvielen veränderlichen. *Journal für die reine und angewandte Mathematik*, 136:210–271.

Santiago Herrera, Caio Corro, and Sylvain Kahane. 2024. Sparse logistic regression with high-order features for automatic grammar rule extraction from treebanks. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15114–15125.

Jaap Jumelet, Leonie Weissweiler, Joakim Nivre, and Arianna Bisazza. 2025. MultiBLiMP 1.0: A massively multilingual benchmark of linguistic minimal pairs.

Mark Kaunisto and Martin Schilk, editors. 2024. *Challenges in Corpus Linguistics*, volume 118 of *Studies in Corpus Linguistics*. John Benjamins Publishing Company.

Geoffrey Leech. 1992. Corpora and theories of linguistic performance. *Directions in corpus linguistics*, 1992:105–122.

Natalia Levshina. 2019. Token-based typology and word order entropy: A study based on universal dependencies. *Linguistic Typology*, 23(3):533–572.

Natalia Levshina, Savithry Namboodiripad, Marc Allassonnière-Tang, Mathew Kramer, Luigi Talamo, Annemarie Verkerk, Sasha Wilmoth, Gabriela Garrido Rodriguez, Timothy Michael Gupton, Evan Kidd, Zoey Liu, Chiara Naccarato, Rachel Nordlinger, Anastasia Panova, and Natalia Stoynova. 2023. Why we need a gradient approach to word order. *Linguistics*, 61(4):825–883.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim

Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20.

Patrick Littell, David R. Mortensen, Ke Lin, Katherine Kairis, Carlisle Turner, and Lori Levin. 2017. URIEL and lang2vec: Representing languages as typological, geographical, and phylogenetic vectors. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 8–14, Valencia, Spain. Association for Computational Linguistics.

Zexi Liu, Jingyi Chai, Xinyu Zhu, Shuo Tang, Rui Ye, Bo Zhang, Lei Bai, and Siheng Chen. 2025. ML-Agent: Reinforcing LLM agents for autonomous machine learning engineering.

Luming Lu, Jiyuan An, Yujie Wang, Liner yang, Cunliang Kong, Zhenghao Liu, Shuo Wang, Haozhe Lin, Mingwei Fang, Yaping Huang, and Erhong Yang. 2024. From text to CQL: Bridging natural language and corpus search engine.

Tony McEnery and Andrew Hardie. 2011. *Corpus Linguistics: Method, Theory and Practice*. Cambridge Textbooks in Linguistics. Cambridge University Press.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajič, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An evergrowing multilingual treebank collection. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4034–4043.

OpenAI. 2025. GPT-5 system card. Technical report, OpenAI. Accessed: 2025-10-22.

John Sinclair. 1991. *Corpus, Concordance, Collocation*. Oxford University Press.

Chandan Singh, John X. Morris, Jyoti Aneja, Alexander Rush, and Jianfeng Gao. 2023. Explaining data patterns in natural language with language models. In *Proceedings of the 6th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 31–55.

Hedvig Skirgård, Hannah J. Haynie, Damián E. Blasi, Harald Hammarström, Jeremy Collins, Jay J. Latarche, Jakob Lesage, Tobias Weber, Alena Witzlack-Makarevich, Sam Passmore, Angela Chira, Luke Maurits, Russell Dinnage, Michael Dunn, Ger Reesink, Ruth Singer, Claire Bowern, Patience Epps, Jane Hill, Outi Vesakoski, Martine Robbeets, Noor Karolin Abbas, Daniel Auer, Nancy A. Bakker, Giulia Barbos, Robert D. Borges, Swintha Danielsen, Luise Dorenbusch, Ella Dorn, John Elliott, Giada Falcone, Jana Fischer, Yustinus Ghanggo Ate, Hannah Gibson, Hans-Philipp Göbel, Jemima A. Goodall, Victoria Gruner, Andrew Harvey, Rebekah Hayes, Leonard Heer, Roberto E. Herrera Miranda, Nataliia Hübler, Biu Huntington-Rainey, Jessica K. Ivani, Marilen Johns, Erika Just, Eri Kashima, Carolina Kipf, Janina V. Klingenberg, Nikita König, Aikaterina Koti, Richard G. A. Kowalik, Olga Krasnoukhova, Nora L.M. Lindvall, Mandy Lorenzen, Hannah Lutzenberger, Tônia R.A. Martins, Celia Mata German, Suzanne van der Meer, Jaime Montoya Samamé, Michael Müller, Saliha Muradoglu, Kelsey Neely, Johanna Nickel, Miina Norvik, Cheryl Akinyi Oluoch, Jesse Peacock, India O.C. Pearey, Naomi Peck, Stephanie Petit, Sören Pieper, Mariana Poblete, Daniel Prestipino, Linda Raabe, Amna Raja, Janis Reimringer, Sydney C. Rey, Julia Rizaew, Eloisa Ruppert, Kim K. Salmon, Jill Sammet, Rhiannon Schembri, Lars Schlabbach, Frederick W.P. Schmidt, Amalia Skilton, Wikaliler Daniel Smith, Hilário de Sousa, Kristin Sverredal, Daniel Valle, Javier Vera, Judith Voß, Tim Witte, Henry Wu, Jingting Ye, Maisie Yong, Tessa Yuditha, Roberto Zariquiey, Robert Forkel, Nicholas Evans, Stephen C. Levinson, Martin Haspelmath, Simon J. Greenhill, Quentin D. Atkinson, and Russell D. Gray. 2023. Grambank reveals the importance of genealogical constraints on linguistic diversity and highlights the impact of language loss. *Science Advances*, 9(16).

Piyapath T. Spencer and Nanthipat Kongborrirak. 2025. Can LLMs help create grammar?: Automating grammar creation for endangered languages with in-context learning. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 10214–10227.

Aarohi Srivastava et al. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Trans. Mach. Learn. Res.*, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th Inter-*

national Conference on Neural Information Processing Systems.

Congying Xia, Chen Xing, Jiangshu Du, Xinyi Yang, Yihao Feng, Ran Xu, Wenpeng Yin, and Caiming Xiong. 2024. FOFO: A benchmark to evaluate LLMs' format-following capability. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 680–699.

Jianwei Yan and Haitao Liu. 2023. Basic word order typology revisited: A crosslinguistic quantitative study based on UD and WALS. *Linguistics Vanguard*, 9(1):73–85.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*.

8. Language Resource References

Matthew S. Dryer and Martin Haspelmath. 2013. *WALS Online (v2020.4)*. Zenodo.

Luka Terčon, Kaja Dobrovoljc, Matej Klemen, Tjaša Arčon, and Marko Robnik-Šikonja. 2026. Corpus-grounded evaluation dataset for grammatical question answering GramQA 1.0. Slovenian language resource repository CLARIN.SI.

Daniel Zeman et al. 2025. Universal dependencies 2.16. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL).

The L2 Network: A CEFR-Aligned Knowledge Graph for Grammar Domain Modeling

Luisa Ribeiro-Flucht, Xiaobin Chen
LEAD Graduate School and Research Network,
University of Tübingen, Germany
{luisa.ribeiro-flucht, xiaobin.chen}@uni-tuebingen.de

Abstract

Large language models have renewed interest in the role of structured linguistic data for applications that require controllable, interpretable, and pedagogically aligned language generation. This need is especially visible in intelligent language tutoring, where grammar cannot be modeled as a flat inventory of patterns alone, but must also capture their relations and functions they realize. We present the L2 Network, a machine-readable knowledge graph of CEFR A1-A2 English grammar that encodes formal patterns, functions, and typed relations between them. The resource is grounded in established pedagogical reference materials, combining form inventory and progression information from the English Grammar Profile with a functional layer derived from CEFR descriptors. We further report content validation of the form-function mappings through expert annotation, including agreement analysis and a consensus-filtered core release. The resulting graph provides an explicit schema for representing pedagogically relevant grammatical knowledge and supports downstream uses such as learner modeling, adaptive task selection, and controlled generation in dialogue-based ICALL systems.

Keywords: knowledge graph, structured linguistic data, CEFR, grammar modeling, ICALL

1. Introduction

Intelligent Computer-Assisted Language Learning (ICALL) has long sought to support second language (L2) learners through scalable and adaptive Intelligent Tutoring Systems (ITS). The recent rise of GenAI, particularly large language models (LLMs), has significantly expanded what is possible. Because LLMs can generate fluent, contextualized, and even multilingual text on demand, they have made it increasingly feasible to move beyond fixed exercise formats toward more open-ended, interactive forms of language practice, as well as to generate personalized content and individualized feedback. Recent work has therefore increasingly treated LLMs as a promising foundation for next-generation ITSs.

More fundamentally, however, this shift raises not only a tutoring challenge, but also a representational one. Language is increasingly understood as a multidimensional system in which formal patterns are associated with semantics, pragmatics, and discourse. For computational applications that require linguistic grounding and controlled generation, language knowledge therefore benefits from being represented as structured data. In ICALL, this need is especially visible: recent work shows that prompting LLMs to generate proficiency-aligned content or to act as adaptive tutors does not reliably produce pedagogically aligned output (Benedetto et al., 2025; Borchers and Shou, 2025; Jokinen, 2024).

Within ITS architectures, domain models provide an explicit representation of what is to be learned (Padayachee et al., 2002). In language tutoring, they have often taken the form of inventories of knowledge components or exercise-linked units, supporting exercise generation, feedback, and learner tracking (Katinskaia, 2025). While such representations can be adequate for relatively fixed exercise formats, they are more limited in open-ended interactive settings, where systems must do more than select target forms: they must also place them in meaningful contexts, guide generation toward relevant realizations, and interpret learner performance across related forms and functions. This motivates representing grammar as structured, relational data rather than as a flat inventory of items.

In this paper, we present the L2 Network as a structured linguistic resource and knowledge graph schema for representing CEFR-aligned English grammar in machine-readable form. The graph formalizes grammatical patterns (e.g., SUBJ have V-en), functions (e.g., describing experiences or expressing degree), and the typed relations that connect them, including realization, dependency, and progression.

Grounded in established pedagogical resources, the L2 Network is introduced both as an English-language resource and as a reusable modeling approach for transforming CEFR-aligned reference materials into structured, machine-readable domain models for other languages. Beyond its immediate value as an explicit domain representation, it also provides a foundation for future integration methods, such as graph-based retrieval, neuro-symbolic control, and graph-informed fine-tuning. The following sections describe its design principles, construction criteria, and initial validation.

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 148–159

2. Background

To clarify what a pedagogically useful language domain model should represent, the present work draws on functional and cognitive traditions, which treat language as a system in which forms are tied to meaning and use (Behrens, 2009; Larsen-Freeman, 2003). Within this broader perspective, usage-based and constructionist approaches (Goldberg, 2003) analyze lexicon and grammar as a continuum of conventionalized form-meaning pairings, or constructions. These range from bound morphemes (e.g., third-person singular -s), to formulaic patterns (e.g., "would you mind ..."), to fully or partially schematic patterns (e.g., the X-er, the Y-er). On this view, linguistic knowledge is best represented as an interconnected network of constructions, a "constructicon" (Diessel, 2023). L2 development, in turn, can be understood as the gradual strengthening of mappings between learners' communicative intentions and the conventionalized forms of the target language that express them (Schmid, 2015).

These perspectives suggest that language is more appropriately modeled as a structured and interconnected system linking formal patterns to meanings and patterns of use, rather than as a flat inventory of isolated items. This raises a representational question: how can such multidimensional knowledge be formalized in a way that supports explicit encoding, interpretation, and computational use? In language education, one influential answer has been the functional-notional tradition, which organizes language around what learners are able to do with it in context rather than around forms alone. This orientation is carried forward in the Common European Framework of Reference for Languages (CEFR, (Council of Europe, 2001)), where proficiency is defined in terms of communicative competence.

In the CEFR, L2 competence is described through language-agnostic can-do statements that specify communicative achievements across proficiency levels. This reflects and extends the Council of Europe's earlier functional-notional tradition, especially Threshold Level 1990 (Van Ek et al., 1991), which organized language in terms of notions and communicative functions. For domain modeling, this is particularly valuable because it provides a principled functional layer: an abstract inventory of communicative goals that can serve as one dimension of a structured representation. Language-specific grammatical realizations can then be linked to this layer as alternative or complementary ways of expressing shared functions.

To support curriculum design and language-specific implementation, Reference Level Descriptions (RLDs)[1] have been developed to specify the linguistic realizations associated with CEFR progression in individual languages. From the perspective of domain modeling, they provide a bridge from abstract proficiency descriptors to language-specific realizations that are suitable for explicit, structured encoding. For English, this bridge is made linguistically explicit in the English Profile program (Harrison and Barker, 2015), including the English Grammar Profile (EGP; O'Keeffe and Mark, 2017) and the English Vocabulary Profile (EVP; Capel, 2012), which organize grammatical and lexical features in relation to CEFR levels.

The EGP, in particular, provides a corpus-informed account of how grammatical features emerge in learner production across CEFR levels. Derived from a four-year quasi-longitudinal analysis of the Cambridge Learner Corpus, it identifies criterial features: linguistic markers that statistically distinguish adjacent proficiency levels through systematic comparison of learner and native-speaker usage. This makes it a valuable resource for grounding domain models in empirically observed relationships between linguistic realizations and proficiency development.

2.1. Graph-based domain modeling

As discussed, functional and usage-based traditions often conceptualize grammar as an interconnected system in which linguistic patterns are related through structural and functional links (Diessel, 2019). Crucially, form-function mappings are rarely one-to-one: a given form may serve multiple functions, and a given function may be realized by multiple forms. This relational organization makes grammar difficult to capture as a flat inventory of isolated items and instead motivates representations that can explicitly encode multiple types of connections.

Knowledge graphs provide such formalism by representing relational knowledge as networks of entities linked through typed relations, thereby supporting explicit organization, traversal and visualization (Peng et al., 2023). In the present case, this means representing grammatical forms, functions, and the relations between them in a machine-readable structure. As illustrated in Figure 1, such a graph can encode not only which forms realize which functions, but also how forms relate to one another through dependency and progression relations. This makes it possible to model the language domain as a structured knowledge space.

More broadly, robust domain models are essential whenever complex knowledge must be repre-

[1] https://www.coe.int/en/web/common-european-framework-reference-languages/reference-level-descriptions

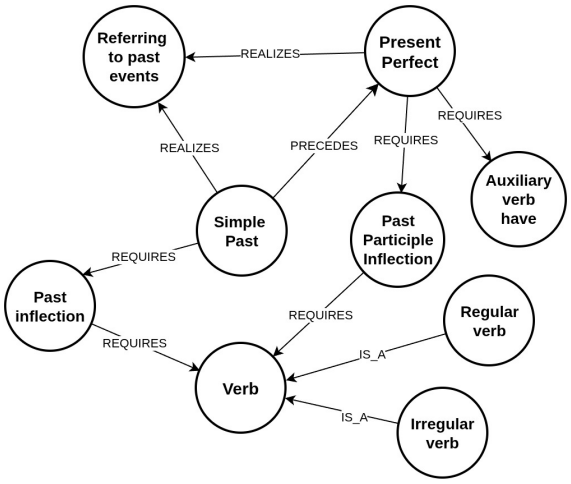


Figure 1: Illustrative example of graph-based grammar modeling. Form and Function nodes are connected through typed relations.

sented in terms of interdependent concepts and realizations (Bagchi, 2021). In exact domains such as mathematics, systems like ALEKS (Cosyn et al., 2021) demonstrate the value of relational domain modeling for adaptive learning. Grounded in Knowledge Space Theory, ALEKS organizes content through explicit prerequisite relations over large sets of topics. By contrast, ICALL has paid comparatively less attention to modeling the internal relational structure of the grammatical domain (Slavuj et al., 2017).

Recent work reflects growing interest in graph-structured approaches, though with different goals. Zhang and Li (2025), for instance, develop a dynamic vocational-English knowledge graph for real-time exercise recommendation. Vu et al. (2025) construct knowledge graphs of Russian grammar prerequisites inferred from learner performance within an ITS and evaluate their alignment with topic difficulty. In parallel, NLP-oriented work has focused more directly on formalizing linguistic knowledge itself, for example by integrating syntax with lexical-semantic networks (Prost, 2022; Rambelli et al., 2019) or by encoding grammatical constraints in ontologies (Aranovich, 2023).

Taken together, these strands point to a gap: existing graph-based approaches are typically not grounded in CEFR-based form-function mappings, do not treat pedagogical reference materials as structured linguistic data to be formalized, and are rarely released as openly reusable domain models. The present work addresses this gap by constructing a pedagogically curated, machine-readable graph of English grammar grounded in established CEFR-aligned resources.

3. The L2 Network

3.1. Implementation Considerations

The L2 Network is a typed knowledge graph that represents grammatical constructions associated with the CEFR Basic User macro-level (A1 and A2). We focus on these entry levels as a principled first step, allowing the resource to be developed and validated on a well-defined portion of the grammar domain before extension to higher proficiency bands.

The graph is implemented in Apache AGE[2], an open-source graph database built on PostgreSQL. This implementation supports the explicit encoding of typed nodes, typed edges, and metadata annotations, while also enabling graph-native querying through Cypher. As a result, the resource can be queried not only as stored data, but also as a traversable linguistic structure, for example to retrieve all forms linked to a given function, identify related or competing constructions, or trace dependency paths across relation types.

The L2 Network is publicly available on GitHub[3] under the CC BY 4.0 license. The repository includes the graph schema, machine-readable exports of nodes and edges with their properties, documentation of node and relation types, and example queries for exploration and reuse. For transparency and reproducibility, it also provides the underlying annotation-derived endorsement data together with the consensus-filtered core resource described in this paper.

3.2. Formalism and Schema

This section presents the explicit schema by which the L2 Network represents grammatical knowledge as structured data. It specifies the node types, relation inventory, and assignment criteria used to convert information from pedagogical reference materials into a typed, machine-readable graph. Formally, the resource is modeled as a directed, labeled graph $G = (V, E)$, where nodes V correspond to linguistic entities and edges E encode typed, directional relations between them. The schema is organized around two core layers, a formal layer and a functional layer, which are connected through explicitly defined relation types. Each node is assigned a unique identifier and CEFR-level metadata, and each edge contributes relational semantics to the overall structure of the domain.

[2] https://age.apache.org/

[3] https://github.com/luisards/l2-network

3.2.1. The Formal Layer (The "What")

The formal layer encodes the grammatical patterns of the domain as an explicit schema of `Form` nodes connected by typed, directed edges. It therefore constitutes the part of the resource that represents language-specific grammatical realizations in machine-readable form. Its empirical basis is the EGP, from which we adopt CEFR level assignments and ordering information. In the EGP, forms are grouped into broader grammatical supercategories, and the A1-A2 subset currently represented in the graph spans a diverse range of them, including adjectives, present clauses, modality, determiners, prepositions, noun phrases, negation, future, passives, adverbs, nouns, conjunctions, pronouns, questions, verbs, and reported speech. These source categories are retained as metadata and support both inspection of the resource and the category-level analyses reported later.

A `Form` node corresponds to an EGP element encoded under the node type *Form*. Depending on the source entry, this may represent a clause-level pattern (e.g., *SUBJ have V-en*), a morphosyntactic pattern (e.g., comparative adjective with *-er*), or a fixed lexical-grammatical unit (e.g., *What a pity!*). Each form is represented as a distinct graph node with an identifier and associated metadata, allowing heterogeneous grammatical units to be captured within a common representational framework.

Each `Form` node contains the following properties:

- `id`: internal identifier
- `label`: human-readable name
- `formula`: abstract structural pattern
- `cefr_level`: CEFR level assignment (A1 or A2)
- `egp_id`: reference to the originating EGP entry
- `example`: example sentence illustrating the form

Forms participate in multiple typed relations, modeled as many-to-many directed edges that define the topology of the formal layer. These relations are intended to make explicit different kinds of linguistic relatedness that are often only implicitly distributed across pedagogical reference materials. In the current version of the graph, three relation types are used which capture distinctions that are especially relevant for structuring the grammar domain: structural dependency, alternative realization, and progression.

Relations & assignment criteria. To make edge assignment explicit and reproducible, each relation type was defined by a separate operational criterion. `PRECEDES` edges were added only when supported by EGP ordering information. `REQUIRES` and `HAS_VARIANT` edges were added only when supported by structural evidence from reference grammars, primarily the *Cambridge Grammar of English* (Carter et al., 2016) and the British Council grammar resource (British Council).

The core relation types are defined as follows:

- `REQUIRES`: encodes structural or compositional dependency between forms. A relation `REQUIRES(A,B)` is assigned when form B cannot be adequately characterized without reference to form A as a necessary structural component. In other words, A contributes an essential part to B, and removing A would change the construction type rather than merely one of its realizations.
- `HAS_VARIANT`: links alternative realizations of a broader grammatical pattern. A relation `HAS_VARIANT(A,B)` is assigned when A and B belong to the same constructional family or instantiate the same broader grammatical pattern, but differ in morphological or structural realization, and neither is a structural prerequisite of the other.
- `PRECEDES`: encodes progression relations between closely related forms. A relation `PRECEDES(A,B)` is assigned when A and B are related constructions and the EGP places A earlier than B in its level-internal ordering, optionally supported by pedagogical grammars that present A as typically introduced before B.

These relations are treated as soft, revisable modeling assumptions rather than absolute claims about acquisition. Their role is to provide an explicit and computationally usable encoding of the internal structure of the English grammar domain, supporting inspection, traversal, and downstream pedagogical applications.

3.2.2. The Function Layer (The "Why")

The function layer encodes the communicative and semantic purposes associated with the form inventory. It provides an abstract layer of representation over language-specific grammatical realizations by modeling the kinds of meanings and discourse purposes that forms may serve. In this sense, it functions as the graph's functional ontology, aligned with the CEFR's functional-notional orientation (Green, 2012).

The function inventory was constructed by surveying CEFR A1 and A2 descriptors (Council of Europe) together with the notions and language functions associated with the relevant constructions in *Threshold Level 1990*. Overlapping or near-equivalent categories were then consolidated into a unified taxonomy, yielding a controlled inventory of function labels for use in the graph (c.f. 9 for the complete list).

Because the forms represented in the graph differ in abstraction, schematicity and lexical specificity, the associated functions also vary in granularity. In the present work, "function" is used as an umbrella term for several types of language-use categories, including communicative functions (e.g., *describing*, *reporting*), grammatical-semantic functions (e.g., *expressing degree*, *referring to entities*), pragmatic functions (e.g., *expressing politeness*, *signaling certainty or uncertainty*), and discourse-organizational functions (e.g., *introducing a topic*, *adding emphasis or contrast*). Each function was assigned a short operational definition to support consistent and inspectable mapping between forms and functions.

Each `Function` node contains the following properties:

- `function_id`: stable internal identifier
- `label`: human-readable function name
- `definition`: short operational description

The interface between the formal and functional layers is encoded through the relation `REALIZES`. A relation `REALIZES(A,B)` links a `Form` node A to a `Function` node B when the form is conventionally associated with that function in the pedagogical domain represented by the graph. The relation is not intended to claim that a form has a single inherent meaning or that the mapping is exhaustive. Rather, it encodes a pedagogically salient and linguistically motivated association between a grammatical realization and a function relevant to A1-A2 instruction and learner modeling. Because forms are often multifunctional and functions may be realized by multiple forms, `REALIZES` is modeled as a many-to-many relation. Where relevant, mappings may be further constrained to a particular CEFR level.

3.3. Structural Characteristics

To characterize the internal topology of the L2 Network, we report descriptive graph statistics capturing node distribution, relational density, and connectivity patterns in Table 1. These metrics provide a quantitative overview of the resource's structural composition and indicate that the graph forms a connected, queryable representation suitable for downstream analysis and computational use.

4. Content Validation

Our validation focused on content coverage and annotation validity, specifically on whether the form-function mappings encoded in the L2 Network adequately capture the functional scope of CEFR A1-A2 grammar. Because the form inventory itself is derived from the English Grammar Profile, the present validation targets the newly introduced functional layer rather than form inclusion. More precisely, the aim was to assess whether the communicative and semantic functions linked to each `Form` node constitute plausible and sufficiently shared interpretations of the grammatical pattern in typical A1-A2 use.

Assigning functions to grammatical forms is inherently qualitative and interpretive. Functions reflect context-sensitive communicative purposes, may overlap conceptually, vary in granularity, and admit multiple plausible analyses (Landert et al., 2023; Weisser, 2014). For this reason, expert review is standard practice in the validation of linguistically interpreted resources. In such settings, agreement metrics are best understood not as a means of eliminating all variation, but as a way of identifying the stable core of shared expert judgment. The validation procedure was therefore designed to preserve interpretive nuance while still supporting systematic resource construction: annotators could assign multiple labels and add free-text suggestions, allowing the study to capture both consensus and principled variation in the proposed mappings.

4.1. Participants and Materials

To evaluate the quality of the form-function mappings, we conducted an expert validation study with 18 annotators with backgrounds in linguistics and language teaching who were either native speakers of English or highly proficient users of English. Each annotator evaluated a set of 40 candidate items, distributed such that every item was reviewed by 2-4 annotators (mean: 2.83). For each item, annotators were asked to indicate which function(s) a given form could realize in typical A1-A2 communication.

Each annotation item included:

- the form label and formula,
- the CEFR level assignment,
- 2-3 illustrative examples, and
- a list of 5 candidate functions.

The candidate set included the target mapping, determined as described in Section 3.2.2, together with function labels drawn from the same or closely related grammatical super-categories. This design allowed us to test not only whether the proposed

Metric	Definition	Value
Total forms	Number of `Form` nodes	314
A1 forms	Number of `Form` nodes with `cefr_level = 'A1'`	119
A2 forms	Number of `Form` nodes with `cefr_level = 'A2'`	195
Functions	Number of `Function` nodes	59
Form Relations	Number of form-to-form edges	1,327
Function Relations	Number of `REALIZES` edges	600
Average degree	Mean number of functions per form (mappings / forms)	1.91
% forms with >1 function	`Form` nodes connected to more than 1 `Function` nodes	69.3%
Orphan check	Forms with neither functions nor grammatical relations	0

Table 1: Structural properties of the L2 Network.

mapping was selected, but also whether nearby alternatives were consistently preferred, thereby providing evidence about the adequacy and distinctiveness of the functional annotation layer.

4.2. Inter-Annotator Agreement

Inter-annotator agreement was assessed using three complementary metrics in order to characterize the stability of the proposed form-function annotation layer:

- **Krippendorff's α.** Because the task is multi-label, agreement was computed at the level of individual label decisions. For each standard function label, presence or absence was treated as a binary nominal variable. Krippendorff's α was used to account for chance agreement and the variable number of annotators per item (2-4).

- **Mean Jaccard similarity.** For each pair of annotators evaluating the same form, Jaccard similarity was computed as $|A \cap B|/|A \cup B|$, measuring overlap between the selected function sets.

- **Exact match rate.** This was defined as the proportion of annotator pairs who selected identical function sets for a form, representing strict set-level agreement.

Across the full annotation matrix, agreement was moderate (Krippendorff's $\alpha = 0.58$). However, aggregate agreement values mask substantial variability across forms, as illustrated in Figure 2. Upon further inspection, disagreement was primarily additive rather than contradictory: annotators typically converged on a shared core function, but differed in whether additional functions should also be assigned. This pattern is consistent with the interpretive nature of functional annotation and suggests that the main source of variation lies in the breadth of mappings retained, rather than in direct opposition over a form's primary function.

4.3. Expert Endorsement Rate

To quantify support for individual form-function mappings, we computed an endorsement rate for each pair, defined as the proportion of experts assigning a given function to a given form. Let n_s denote the number of experts selecting the function and n the total number of experts evaluating that item. The endorsement rate r is given by:

$$r = \frac{n_s}{n}$$

Endorsement rates were computed for every candidate form-function pair in order to examine the distribution of consensus across the annotation layer. This allows us to distinguish highly stable mappings from those that attract weaker or more divided support, and to better understand whether disagreement reflects direct contradiction or the addition of secondary functions.

Endorsement band	Vote patterns	Mappings
Unanimous	$2/2,\ 3/3,\ 4/4$	309
75% annotators	$3/4$	18
67% annotators	$2/3$	76
50% annotators	$2/4,\ 1/2$	133
Below 50%	$1/3,\ 1/4$	242

Table 2: Distribution of endorsement levels across candidate form-function pairs by attainable vote pattern.

As shown in Table 2, most candidate pairs received either unanimous endorsement or majority support, while a substantial minority fell below the 50% level. In line with the agreement analysis above, these lower-consensus cases often reflect additive disagreement: annotators typically converged on a shared core function, but differed in whether additional functions should also be assigned. Endorsement rates therefore provide a useful descriptive view of which parts of the functional layer are especially stable and which remain more weakly supported.

145 forms achieved both monofunctionality and unanimous endorsement, representing the highest-confidence mappings in the resource. These

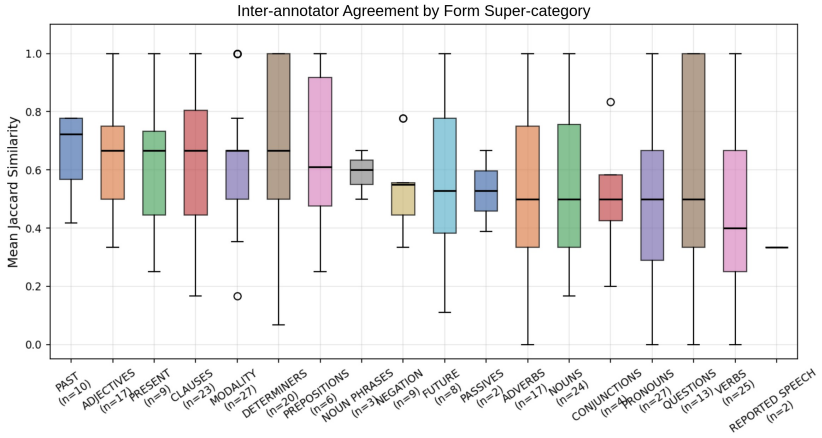


Figure 2: Distribution of inter-annotator agreement across forms' super-categories.

cases can be interpreted as especially stable form-function associations within the current annotation scheme. Examples are shown in Table 3.

4.4. Consensus Analysis and Resource Release

Because the number of annotators per form varied between two and four, the same endorsement percentage does not always correspond to the same evidentiary standard. To derive a stable core resource for public release, we therefore adopted an *independent corroboration* criterion: a mapping is retained only if it was endorsed by at least two annotators ($n_{\text{sel}} \geq 2$).

This criterion applies a consistent principle across the dataset: each released mapping must be supported by agreement between at least two independent judges, even though the equivalent endorsement rate differs by annotator group size (Table 4).

Applying this filter yielded 429 mappings, reducing the original inventory by 171 mappings (28.5%) and 2 forms. The two excluded forms, *Pronoun subject it for first person* and *Verb infinitive form*, were cases in which annotators selected entirely different functions, indicating that these entries require further adjudication. We therefore release both the full endorsement data and the independently corroborated subset, preserving lower-consensus mappings for future analysis while providing a principled consensus-based core annotation layer for reuse.

4.5. Operational Integration

To demonstrate runtime usability, we integrated the L2 Network into AISLA (Chen et al., 2022), a task-based dialogue system for curriculum-aligned English practice. Within this architecture, the L2 Network functions as a queryable domain model that determines which forms can be selected, how they map to functions, and under what constraints they are instantiated.

4.6. Implementing a Contextual Layer (The "How")

While the formal and functional layers represent what linguistic knowledge is modeled, instructional systems must also specify how that knowledge is practiced. In task-based language learning, AISLA's pedagogical framework, forms are practiced through interactional situations in which particular form-function pairings become meaningful and pragmatically licensed. To capture this contextual layer, we introduce a `Task` node type connected to `Form` and `Function` nodes via the typed edges `TARGETS_FORM` and `TARGETS_FUNCTION`. An example of these layers is shown in Figure 3.

Each `Task` node represents a parameterized dialogue template grounded in a common real-life interactional scenario that pragmatically licenses one or more target forms and functions. In this way, the graph does not only represent what linguistic knowledge is modeled, but also how that knowledge can be contextualized for instruction. Each `Task` node contains the following properties:

- `id`
- `task_name`
- `form_id`
- `function_id`
- `system_instructions`
- `student_instructions`

Form	Formula	Function
AffirmativeDeclarativeClause	`SUBJ V (...)`	Making statements
AffirmativeInterrogativeClauseWithBe	`(wh) BE SUBJ (...)`	Asking for information
AttributiveAdjective	`ADJ N`	Describing
FutureWillAffirmative	`SUBJ will V`	Referring to future actions and events
QuantifierWithPluralNoun	`QUANT N.PL`	Expressing quantity
DeterminerWithNoun	`DET N`	Referring to entities
TimeAdverbEnd	`CLAUSE ADV.TIME`	Situating events in time
BeSureClause	`be sure that CLAUSE`	Expressing certainty

Table 3: Examples of unanimously endorsed, single-function mappings.

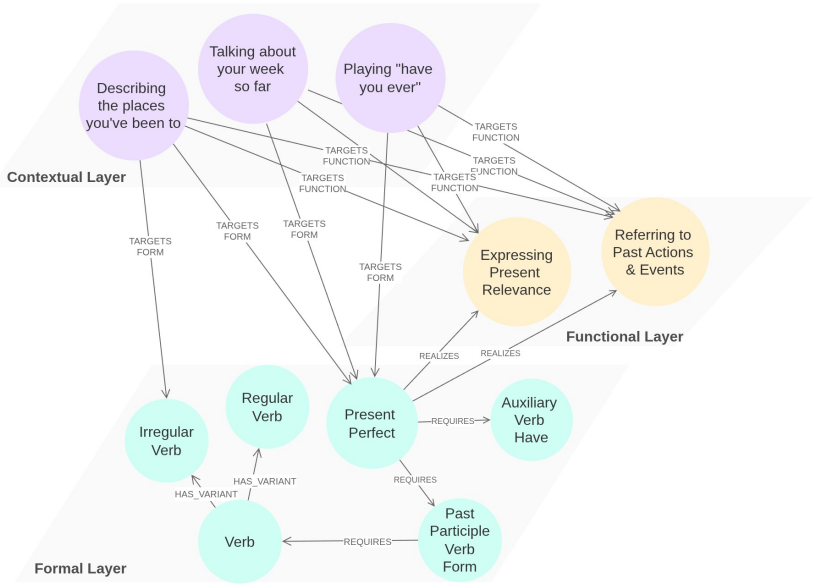


Figure 3: Example of the contextual extension layer in the L2 Network. Task nodes connect pedagogical contexts to the underlying function and form layers.

Annotators	Min. support ($n_{\text{sel}} \geq 2$)	Rate
2	$2/2$	100%
3	$2/3$	67%
4	$2/4$	50%

Table 4: Equivalent endorsement rates for the independent corroboration criterion across annotator group sizes.

Because task templates are linked to forms and functions through graph relations rather than hard-coded directly into prompts, linguistic targeting and contextual realization can be handled separately. This makes it possible to vary instructional contexts while preserving explicit control over what is being practiced.

The graph is particularly relevant for learner modeling because it provides an explicit structure over which a learner overlay can be defined. Rather than tracking performance only on isolated items, the system can associate learner evidence with `Form` and `Function` nodes and interpret this evidence in relation to neighboring nodes. This makes it possible to represent learner state not simply as a list of correct or incorrect targets, but as a structured profile over the grammar domain.

Such an overlay can support more principled adaptive decisions. Depending on the learner's observed performance, the system may identify forms or functions that appear underdeveloped, revisit prerequisite structures, remain within the same functional area while shifting to a related form, or select practice that reinforces a weak form-function mapping. In this way, the graph supports decisions about what content should be delivered next. Rather than relying on an LLM to infer pedagogical priorities, the the addition of the graph makes the relevant domain structure explicit and queryable.

5. Conclusion & Future Work

This paper introduced the L2 Network, a graph-based English grammar domain model that links CEFR-aligned grammatical forms to functions via explicit relations. By making linguistic targets and their dependencies machine-readable, this re-

source provides a transparent substrate for principled sequencing, learner-model overlays, and integration with LLM-based generation. A central advantage of the proposed schema is its extensibility. It distinguishes a stable grammatical core from additional system layers, enabling platform-specific adaptation without modifying the underlying form-function mappings.

The analysis indicates meaningful expert convergence on the functions of Basic User-level forms, while also reflecting the inherently interpretive nature of form-function mapping. The observed Krippendorff's α of 0.58 suggests a shared but still evolving annotation space rather than a fully stabilized inventory. At the same time, the substantial proportion of unanimously endorsed mappings indicates that the current version of the graph already captures a stable core of form-function associations. We therefore treat these results as encouraging initial validity evidence, while recognizing that the resource remains a work in progress and will require continued refinement and evaluation.

Several directions follow: First, we plan to extend the graph with lexical realizations, supporting more reliable controlled text generation by conditioning LLM outputs on explicit specifications. Second, future work will extend the present content-validation study toward predictive and external criterion validation. In particular, we will examine whether graph-defined form-function mappings and level assignments predict learner production patterns and CEFR-aligned proficiency outcomes in independent datasets.

Finally, although instantiated here for English, the design is intended to be transferable across languages. The CEFR provides a shared, language-agnostic functional backbone, while RLDs supply language-specific realizations at each level. For languages with comparable RLD resources, the L2 Network offers a generalizable blueprint for constructing structured, data-informed domain models that can support transparent adaptivity in ICALL.

Beyond domain modeling, the L2 Network provides a symbolic substrate for neuro-symbolic approaches to GenAI, curriculum and syllabus design as well as assessment design and test coverage analysis. Because constructions (forms-function pairings) and their dependencies are represented explicitly, the graph can support symbolic planning of pedagogical intent (e.g., selecting function-consistent targets under soft prerequisite and competition constraints) followed by neural realization via LLM generation. It also enables constraint- and verification-based workflows in which generated dialogues are automatically checked against graph-derived expectations and iteratively refined. Furthermore, graph neighborhoods can be retrieved as structured context to condition generation (structure-aware retrieval), or translated into soft or hard decoding constraints to improve controllability.

6. Acknowledgements

We thank the researchers who participated in the expert survey and contributed their time and expertise to the validation of the function mappings. Their feedback was essential for assessing the clarity and consistency of the proposed annotations.

We also acknowledge the use of the *English Grammar Profile* as the primary resource for the form inventory and CEFR-level anchoring used in this work.

This work was supported by the Hector Foundation (Pillar III).

7. Limitations

A primary limitation of the current resource lies in the compression of annotator judgments into a filtered consensus graph. While this step improves consistency and yields a more stable set of mappings for downstream implementation, it necessarily suppresses minority selections that may reflect context-sensitive or more fine-grained functional interpretations. The filtered inventory is therefore likely more robust for modeling and pedagogical control, but less exhaustive as a representation of the full functional variability associated with the constructions.

A second limitation concerns scope. The current version of the L2 Network covers only English grammar at the CEFR Basic User levels (A1-A2). Although this restriction was intentional in order to support careful schema development and validation, it means that the present resource does not yet capture higher proficiency levels, broader discourse phenomena, or cross-linguistic variation.

A further limitation is that several aspects of the graph depend on expert modeling decisions. This applies not only to the assignment of form-function mappings, but also to the granularity of the function inventory and to the operational criteria used to define relations. These choices were made transparently and grounded in established reference materials, but they remain revisable as the resource evolves.

Finally, the validation reported here is limited to content coverage, agreement patterns, and structural consistency. Although these provide important initial evidence for the usefulness of the resource, they do not by themselves establish the effectiveness of the graph in downstream applications such as learner modeling, adaptive task selection, or controlled generation. Such uses require separate empirical evaluation.

8. Bibliographical References

Raúl Aranovich. 2023. Modeling grammars with knowledge representation methods: Subcategorization as a test case. In *European Semantic Web Conference*, pages 112–116. Springer.

Mayukh Bagchi. 2021. A large scale, knowledge intensive domain development methodology. *Knowledge Organization: KO*, 48(1):8.

Heike Behrens. 2009. Usage-based and emergentist approaches to language acquisition.

Luca Benedetto, Gabrielle Gaudeau, Andrew Caines, and Paula Buttery. 2025. Assessing how accurately large language models encode and apply the common european framework of reference for languages. *Computers and Education: Artificial Intelligence*, 8:100353.

Conrad Borchers and Tianze Shou. 2025. Can large language models match tutoring system adaptivity? a benchmarking study. In *International Conference on Artificial Intelligence in Education*, pages 407–420. Springer.

British Council. Grammar. LearnEnglish. Accessed: 2026-02-17.

Annette Capel. 2012. Completing the english vocabulary profile: C1 and c2 vocabulary. *English Profile Journal*, 3:e1.

Ronald Carter, Michael McCarthy, Geraldine Mark, and Anne O'Keeffe. 2016. English grammar today: An a–z of spoken and written grammar. Web adaptation in Cambridge Dictionary. Accessed: 2026-02-17.

Xiaobin Chen, Elizabeth Bear, Bronson Hui, Haemanth Santhi-Ponnusamy, and Detmar Meurers. 2022. Education theories and ai affordances: Design and implementation of an intelligent computer assisted language learning system. In *International Conference on Artificial Intelligence in Education*, pages 582–585. Springer.

Eric Cosyn, Hasan Uzun, Christopher Doble, and Jeffrey Matayoshi. 2021. A practical perspective on knowledge space theory: Aleks and its data. *Journal of Mathematical Psychology*, 101:102512.

Council of Europe. 2001. *Common European framework of reference for languages: Learning, teaching, assessment*. Cambridge University Press.

Council of Europe. 2020. *Common European Framework of Reference for Languages: Learning, teaching, assessment – Companion Volume*. Cambridge University Press, Cambridge.

Holger Diessel. 2019. *The Grammar Network*. Cambridge University Press.

Holger Diessel. 2023. *The constructicon: Taxonomies and networks*. Cambridge University Press.

Adele E Goldberg. 2003. Constructions: A new theoretical approach to language. *Trends in cognitive sciences*, 7(5):219–224.

Anthony Green. 2012. *Language functions revisited: Theoretical and empirical bases for language construct definition across the ability range*, volume 2. Cambridge University Press.

Julia Harrison and Fiona Barker. 2015. *English profile in practice*, volume 5. Cambridge University Press.

Kristiina Jokinen. 2024. The need for grounding in llm-based dialogue systems. In *Proceedings of the Workshop: Bridging Neurons and Symbols for Natural Language Processing and Knowledge Graphs Reasoning (NeusymBridge)@ LREC-COLING-2024*, pages 45–52.

Anisia Katinskaia. 2025. An overview of artificial intelligence in computer-assisted language learning. *arXiv preprint arXiv:2505.02032*.

Daniela Landert, Daria Dayter, Thomas C. Messerli, and Miriam A. Locher. 2023. *Corpus Pragmatics*. Cambridge Elements in Pragmatics. Cambridge University Press.

Diane Larsen-Freeman. 2003. Teaching language: From grammar to grammaring. *(Heinle Cengage Learning)*.

Anne O'Keeffe and Geraldine Mark. 2017. The english grammar profile of learner competence: Methodology and key findings. *International Journal of Corpus Linguistics*, 22(4):457–489.

Indira Padayachee et al. 2002. Intelligent tutoring systems: Architecture and characteristics. In *Proceedings of the 32nd Annual SACLA Conference*, pages 1–8. South African Computer Lecturers' Association.

Ciyuan Peng, Feng Xia, Mehdi Naseriparsa, and Francesco Osborne. 2023. Knowledge graphs: Opportunities and challenges. *Artificial intelligence review*, 56(11):13071–13102.

Jean-Philippe Prost. 2022. Integrating a phrase structure corpus grammar and a lexical-semantic network: the holinet knowledge graph. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 613–622.

Giulia Rambelli, Emmanuele Chersoni, Philippe Blache, Chu-Ren Huang, and Alessandro Lenci. 2019. Distributional semantics meets construction grammar. towards a unified usage-based model of grammar and meaning. In *Proceedings of the First International Workshop on Designing Meaning Representations*, pages 110–120.

Hans-Jörg Schmid. 2015. A blueprint of the entrenchment-and-conventionalization model. *Yearbook of the German cognitive linguistics association*, 3(1).

Vanja Slavuj, Ana Meštrović, and Božidar Kovačić. 2017. Adaptivity in educational systems for language learning: a review. *Computer Assisted Language Learning*, 30(1-2):64–90.

Jan Ate Van Ek, John Leslie Melville Trim, et al. 1991. *Threshold level 1990*. Council of Europe.

Anh-Duc Vu, Jue Hou, Anisia Katinskaia, Ching-Fan Sheu, and Roman Yangarber. 2025. A bayesian approach to inferring prerequisite structures and topic difficulty in language learning. In *BEA: Proceedings of the 20th Workshop on Innovative Use of NLP for Building Educational Applications*.

Martin Weisser. 2014. *Speech act annotation*, page 84–114. Cambridge University Press.

Fengna Zhang and Xinxin Li. 2025. Dynamic recommendation system for higher vocational english learning paths based on real-time knowledge graph update. *Engineering Reports*, 7(12):e70492.

9. Appendix

ID	Function	ID	Function
1	Asking for advice	31	Giving advice
2	Asking for confirmation	32	Giving directions or instructions
3	Asking for information	33	Making offers and invitations
4	Comparing	34	Making suggestions
5	Comparing within a set	35	Providing detail
6	Conveying meaning	36	Referring back to entities
7	Making statements	37	Referring to actions states or relations
8	Describing	38	Referring to additional or alternative entities
9	Emphasizing the affected entity	39	Referring to entities
10	Expressing obligation or necessity	40	Referring to future actions and events
11	Expressing beliefs and opinions	41	Referring to impersonal events
12	Expressing cause	42	Referring to past actions and events
13	Expressing certainty	43	Referring to past events with present relevance
14	Expressing condition	44	Referring to present states events and facts
15	Expressing degree	45	Referring to self
16	Expressing evaluation	46	Referring to self-directed actions
17	Expressing frequency	47	Referring to the acting or experiencing entity
18	Expressing hypotheses or inferences	48	Referring to the affected entity
19	Expressing manner	49	Referring to third-person subjects
20	Expressing negation	50	Reporting information
21	Expressing politeness	51	Requesting
22	Expressing possession	52	Sequencing events
23	Expressing possibility	53	Situating events or entities
24	Expressing ability	54	Situating entities in space
25	Expressing preference	55	Situating events in time
26	Expressing probability	56	Specifying information
27	Expressing progression	57	Pointing to entities
28	Expressing quantity	58	Expressing contrast
29	Expressing wishes	59	Expressing indefiniteness
30	Framing or linking information		

Table 5: Inventory of communicative functions ($n = 59$).

Paraphrase Acquisition via Bilingual Pivoting Based on Neural Word Alignment

Risa Kondo[†] **Seiji Sugiyama**[†] **Tomoyuki Kajiwara**[†‡] **Takashi Ninomiya**[†]
[†] Graduate School of Science and Engineering, Ehime University, Japan
[‡] D3 Center, The University of Osaka, Japan
{kondo@ai.cs., sugiyama@ai.cs., kajiwara@cs., ninomiya.takashi.mk@}ehime-u.ac.jp

Abstract

We utilize neural word alignment to improve the quality of paraphrase databases in English and Japanese. For large-scale paraphrase acquisition, previous studies have employed a framework of bilingual pivoting based on word alignment on bilingual parallel corpora. Naturally, the quality of paraphrases acquired by bilingual pivoting depends on the performance of word alignment. Previous studies based on statistical word alignment have limitations in the quality of acquired paraphrases because they do not consider word meaning. This study employs a more sophisticated neural approach for word alignment in bilingual pivoting to enhance the quality of paraphrase acquisition. Experimental results revealed that our paraphrase databases outperformed existing ones in both internal and external evaluations.

Keywords: Paraphrase Acquisition, Bilingual Pivoting, Neural Word Alignment

1. Introduction

Paraphrase databases have been utilized for various natural language processing tasks, including information retrieval (Liu et al., 2004) and machine translation (Callison-Burch et al., 2006). Even in recent years, when deep learning dominates, paraphrase databases are utilized for tasks such as representation learning (Faruqui et al., 2015; Wieting et al., 2016a,b; Sugiyama et al., 2025), pre-training (Sun et al., 2023) and fine-tuning (Zetsu et al., 2022) of masked language models, and lexical simplification (Pavlick and Callison-Burch, 2016; Nishihara and Kajiwara, 2020; Qiang et al., 2021).

For large-scale paraphrase acquisition, a framework of bilingual pivoting (Bannard and Callison-Burch, 2005) based on word alignment on bilingual parallel corpora has been widely employed. As shown in Figure 1, bilingual pivoting first estimates word alignments on a given parallel corpus. It then extracts phrase pairs in the target language ("first author" and "lead author") corresponding to common phrases in the pivot language ("筆頭著者"), treating them as paraphrases. Naturally, the quality of paraphrases acquired by bilingual pivoting depends on the performance of word alignment. Existing paraphrase databases (Ganitkevitch et al., 2013; Ganitkevitch and Callison-Burch, 2014; Pavlick et al., 2015; Mizukami et al., 2014) are based on statistical word alignment (Och and Ney, 2003) that disregards word meaning, resulting in limited paraphrase quality.

In this study, we employ a more sophisticated neural approach for word alignment in bilingual pivoting to construct higher-quality paraphrase databases.[1] However, while neural word alignment outperforms statistical word alignment, it is incompatible with traditional phrase extraction heuristics designed for statistical word alignment, generating too many phrase alignments in step 2 of Figure 1. To address this challenge, we perform both filtering of phrase pairs and reranking of paraphrase pairs, as shown in Figure 2, to acquire high-quality paraphrases. Experimental results on internal evaluation in Japanese and external evaluation in both Japanese and English reveal the usefulness of our paraphrase databases. We will release[1] both 70 million paraphrase pairs in Japanese and 40 million paraphrase pairs in English.

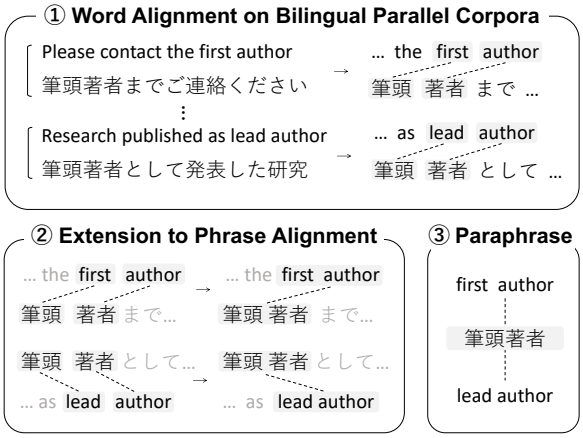


Figure 1: Paraphrase acquisition via bilingual pivoting.

[1] https://github.com/EhimeNLP/EhiMerPPDB

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 160–168
May 11, 2026.

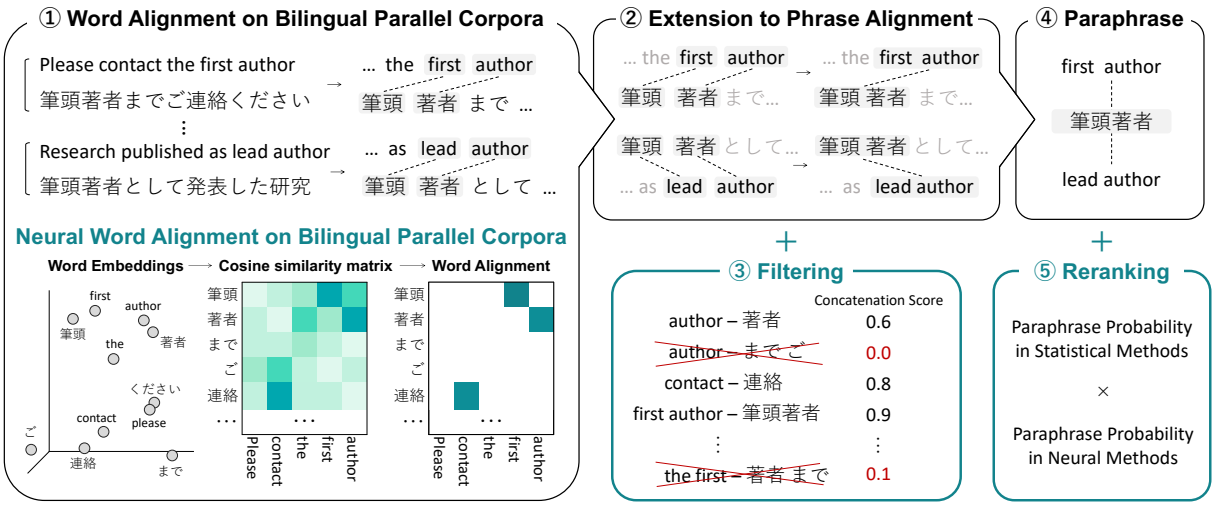


Figure 2: Overview of our proposed method. For bilingual pivoting, we employ a neural approach for word alignment. When extending this to phrase alignment, we reduce noise by filtering phrase pairs. Finally, we combine the paraphrase probability based on statistical word alignment with that of neural word alignment to rerank the paraphrases.

2. Related Work

2.1. Bilingual Pivoting

As shown in Figure 1, bilingual pivoting (Bannard and Callison-Burch, 2005) performs word alignment on bilingual parallel corpora and extends it to phrase alignment to acquire paraphrases. Paraphrases acquired in this way are phrase pairs in the target language that correspond to common phrases in the pivot language. As shown in Equation (1), its paraphrase probability $p(e_2|e_1)$ is estimated by marginalizing over f the translation probability $p(f|e_1)$ from phrase e_1 in the target language to phrase f in the pivot language and the translation probability $p(e_2|f)$ from f to phrase e_2 in the target language.

$$p(e_2|e_1) = \sum_f p(e_2|f)p(f|e_1) \quad (1)$$

Bilingual pivoting has been employed in the construction of large-scale language resources such as PPDB (**p**ara**p**hrase **d**ata**b**ase) in English (Ganitkevitch et al., 2013; Ganitkevitch and Callison-Burch, 2014; Pavlick et al., 2015) and its Japanese versions. One of the previous studies in Japanese, JPPDB[2] (Mizukami et al., 2014), combined several small-scale Japanese-English parallel corpora and acquired paraphrases using bilingual pivoting based on GIZA++ (Och and Ney, 2003), a statistical word alignment. Another previous study, EhiMerPPDB,[3] utilized JParaCrawl[4] (Morishita et al., 2020, 2022), a large-scale Japanese-English parallel corpus, and similarly acquired paraphrases via bilingual pivoting based on GIZA++.

Since these previous studies rely on statistical word alignment (Och and Ney, 2003), they do not reflect information about word meaning and context. In contrast, we employ neural word alignment methods based on masked language models to enhance the quality of paraphrase acquisition.

2.2. Neural Word Alignment

Inspired by pre-trained masked language models such as BERT (Devlin et al., 2019), neural word alignment methods (Jalili Sabet et al., 2020; Azadi et al., 2023) that estimate word alignments from contextual word embeddings are being actively studied. These methods achieve high performance despite not requiring gold alignments for training.

A pioneer in neural word alignment, SimAlign (Jalili Sabet et al., 2020), employs multilingual masked language models such as mBERT[5] (Devlin et al., 2019) and XLM-R[6] (Conneau et al., 2020) to perform word alignment that maximizes word similarity based on contextual word embeddings for sentence pairs. PMIAlign (Azadi et al., 2023)

[2] http://ahcweb01.naist.jp/old/resource/jppdb/

[3] https://github.com/EhimeNLP/EhiMerPPDB

[4] https://www.kecl.ntt.co.jp/icl/lirg/jparacrawl/

[5] https://huggingface.co/google-bert/bert-base-multilingual-cased

[6] https://huggingface.co/FacebookAI/xlm-roberta-base

improves similarity calculations in SimAlign with pointwise mutual information.

In this study, we employ these neural word alignment methods to improve paraphrase acquisition.

3. Paraphrase Acquisition

In this study, we apply bilingual pivoting (Bannard and Callison-Burch, 2005) based on neural word alignment (Jalili Sabet et al., 2020; Azadi et al., 2023) to a Japanese-English parallel corpus to enhance the quality of paraphrase databases in English and Japanese.

Preprocessing for Parallel Corpus As with the EhiMerPPDB,[3] we also used JParaCrawl (Morishita et al., 2020, 2022), the largest Japanese-English parallel corpus. For tokenization, we used Moses Tokenizer (Koehn et al., 2007) for English and MeCab with IPADIC (Kudo et al., 2004) for Japanese. We then filtered out[7] sentence pairs containing blank lines, unnecessary spaces, or longer sentences exceeding 100 words.

Word Alignment We employed SimAlign (Jalili Sabet et al., 2020) and PMIAlign (Azadi et al., 2023) for neural word alignment. These are based on multilingual masked language models, and we employ either mBERT[5] (Devlin et al., 2019) or XLM-R[6] (Conneau et al., 2020). To select a masked language model suitable for neural word alignment, we conducted a preliminary experiment comparing word alignment performance on the KFTT[8] Japanese-English parallel corpus. Evaluation results in Table 1 revealed the effectiveness of mBERT in neural word alignment. Therefore, we employ mBERT-based neural word alignment for paraphrase acquisition.

Extension to Phrase Alignment To extend the word-level alignments obtained on JParaCrawl to phrase-level alignments, we employed the phrase extraction heuristic, which corresponds to Step 5 of the Moses toolkit (Koehn et al., 2007). This process extracts candidate phrases by identifying contiguous sequences of words—including those not explicitly aligned—that are consistent with the word alignments produced in the previous step. Following a previous study (Mizukami et al., 2014), the maximum length of phrases was set to 7 tokens.

Filtering Phrases To ensure the quality of the paraphrase database, we filtered out phrase pairs containing noise. Specifically, we removed phrases that consisted solely of symbols (e.g., punctuation marks or mathematical signs) or did not contain any lexical characters from the target language, as such pairs are unlikely to be meaningful paraphrases. Furthermore, since phrases with low co-occurrence probabilities are often noisy, we filtered them using the phrase extraction tool[9] included in word2vec (Mikolov et al., 2013). This tool scores how likely words are to appear consecutively. To systematically eliminate unreliable candidates, we performed this scoring and phrase formation process iteratively four times for both languages, removing phrases with scores below 200, 100, 50, and 25 in each respective step.

	#Align	Recall	Precision	F1
GIZA++	19,425	0.502	0.367	0.424
SimAlign (mBERT)	9,671	**0.553**	**0.669**	**0.606**
SimAlign (XLM-R)	8,930	0.475	0.614	0.535
PMIAlign (mBERT)	8,827	**0.547**	**0.721**	**0.622**
PMIAlign (XLM-R)	9,025	0.508	0.636	0.565

Table 1: Preliminary experiment: performance of word alignment on the KFTT evaluation dataset.

Reranking Paraphrases Having obtained the phrase pairs and their translation probabilities, we calculated the paraphrase probability using Equation (1). The phrase pairs and their paraphrase probabilities will be added to our paraphrase database. Here, previous study on lexical paraphrase acquisition (Kajiwara et al., 2017) has reported the effectiveness of combining statistical-based and embedding-based methods. Also in this study, the paraphrase probability based on statistical word alignment and the paraphrase probability based on neural word alignment are complementary, and we expect to acquire high-quality paraphrases by combining them. To obtain high-quality paraphrases, we multiplied the paraphrase probabilities of both methods. Since this multiplication takes the intersection of the paraphrase pairs acquired by each method, it reduces the total number of candidates while prioritizing those identified as reliable by both statistical and neural alignments. Here, statistical word alignment using GIZA++ (Och and Ney, 2003) was bidirectionalized with a grow-diag-final heuristic after calculating word alignments for each language direction based on the IBM model 2. Table 2 shows the quantities of paraphrase pairs we acquired.

[7] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/training/clean-corpus-n.perl

[8] https://www.phontron.com/kftt/

[9] https://github.com/tmikolov/word2vec/blob/master/word2phrase.c

		Number of paraphrases
English	PPDB 2.0 (XXXL)	56M
	EhiMerPPDB	251M
	SimAlign	28,176M
	PMIAlign	28,274M
	SimAlign x GIZA++	37M
	PMIAlign x GIZA++	38M
Japanese	JPPDB (10best)	15M
	EhiMerPPDB	387M
	SimAlign	37,231M
	PMIAlign	40,210M
	SimAlign x GIZA++	69M
	PMIAlign x GIZA++	71M

Table 2: Number of paraphrase pairs.

4. Experiments

We experimentally evaluate the quality and usefulness of paraphrase databases both internally, using recall and precision, and externally, on sentence pair modeling tasks. For comparison, we use PPDB 2.0 (Pavlick et al., 2015) in English, JPPDB (Mizukami et al., 2014) in Japanese, and EhiMerPPDB[3] in both languages. EhiMerPPDB is a paraphrase database constructed from JParaCrawl (Morishita et al., 2020, 2022) like ours, but based on statistical word alignment unlike ours.

4.1. Internal Evaluation

In this section, we evaluate the quality of paraphrase databases using recall (automatic evaluation) and precision (human evaluation). However, since PPDB 2.0 is currently private and inaccessible, we report only the experimental results in Japanese.

4.1.1. Setting

Dataset We constructed an evaluation dataset by extracting phrasal paraphrase pairs from a manually built Japanese paraphrase corpus created by experts. To avoid bias toward specific domains and ensure high-quality annotations, we employed three corpora created by different groups of experts: JADES[10] (Hayakawa et al., 2022) for news domains created by a researcher in Japanese text simplification, MATCHA[11] (Miyata et al., 2024) for tourism domains created by Japanese language teachers, and JASMINE[12] (Horiguchi et al., 2024) for medical domains created by annotators with expertise in Japanese medical NLP.[13]

We first randomly selected 200 sentence pairs from each corpus. These pairs were then distributed among three university students, all native Japanese speakers, who manually extracted phrasal paraphrases. The annotators were instructed to identify the minimum units that correspond as paraphrases among the segments tokenized by MeCab (Kudo et al., 2004). As a preprocessing step, similar to Section 3, we removed noise such as phrases containing symbols or those not containing Japanese. Finally, we collected a total of 1,776 Japanese phrasal paraphrases: 895 from JADES, 452 from MATCHA, and 429 from JASMINE.

Recall We automatically evaluated the coverage of paraphrase databases. For a fair comparison with JPPDB, our paraphrase database also evaluated the top 10 entries with the highest paraphrase probability for each query.

Precision We manually evaluated the quality of paraphrase databases. Three university students who are native Japanese speakers evaluated the top 10 paraphrases for 100 randomly selected phrases using the following four-point scale.

1. Not semantically equivalent.
2. Semantically equivalent but not substitutable.
3. Substitutable depending on context.
4. Always substitutable.

We defined paraphrases rated as 1 or 2 as negative examples and those rated as 3 or 4 as positive examples, then calculated the precision based on the majority vote of the evaluators. Here, the inter-annotator agreement was evaluated using the weighted kappa coefficient, showing substantial agreement ranging from 0.56 to 0.72.

4.1.2. Result

Table 3 shows the experimental results. Across all evaluation metrics, the proposed method combining statistical word alignment and neural word alignment (PMIAlign x GIZA++) consistently achieved the highest performance. Neural word alignments collect an extremely large number of paraphrases, and may contain a lot of noise when used on its own. The combination of statistical word alignment and neural word alignment is promising for removing such noise, resulting in the acquisition of a high-quality paraphrase database.

[10] https://github.com/naist-nlp/jades
[11] https://github.com/EhimeNLP/matcha
[12] https://github.com/EhimeNLP/JASMINE

[13] We excluded corpora created via crowdsourcing platforms such as SNOW (Katsuta and Yamamoto, 2018) and corpora generated by machine translation systems such as TMUP (Suzuki et al., 2017).

	Alignment	#Paraphrase	Recall	Precision	F1 Score
JPPDB[2]	GIZA++	15M	0.172	0.379	0.236
EhiMerPPDB[3]	GIZA++	387M	**0.209**	0.439	0.283
Ours	SimAlign	37,231M	0.186	0.344	0.241
Ours	PMIAlign	40,210M	0.191	0.380	0.255
Ours	SimAlign x GIZA++	69M	0.207	0.450	0.284
Ours	PMIAlign x GIZA++	71M	**0.209**	**0.466**	**0.288**

Table 3: Internal evaluation of Japanese paraphrase databases.

	English			Japanese		
Rank	GIZA++	PMIAlign	GIZA++×PMIAlign	GIZA++	PMIAlign	GIZA++×PMIAlign
1	spooky	eerie	spooky	不気味	不気味	不気味
2	eerie	spooky	eerie	恐ろしい	不吉な	恐ろしい
3	uncanny	an eerie	uncanny	キモ	恐ろしい	な
4	weird	uncanny	weird	な	不思議	奇妙な
5	eerily	weird	an eerie	奇妙な	ような	不吉な
6	scary	eerily	eerily	creepy 気味悪い	奇妙な	キモ
7	macabre	ominous	scary	気味悪い	奇妙	気味悪い
8	disgusting	sinister	macabre	で不気味	不思議な	奇妙
9	kimonos	scary	strange	奇妙	気味悪い	不思議な
10	strange	bad	disgusting	おかしい	な	おかしい

Table 4: Top 10 paraphrases for the expression "creepy" / "不気味な"

	Train	Dev	Test
Shopping Queries	1,254,438	138,625	425,762
STS-B	5,749	1,500	1,379
SICK	4,439	495	4,906
SNLI	549,367	9,842	9,824
PAWS	49,401	8,000	8,000

Table 5: Number of sentence pairs in English.

	Train	Dev	Test
Shopping Queries	294,874	32,272	118,907
JSTS	11,205	1,246	1,457
JSICK	4,500	500	4,927
JNLI	18,065	2,008	2,434
PAWS-X	49,401	2,000	2,000

Table 6: Number of sentence pairs in Japanese.

Among the neural word alignment methods, PMIAlign consistently outperformed SimAlign. As shown in Table 1, PMIAlign demonstrates higher performance in word alignment. These consistent results suggest that employing high-performance word alignment methods can improve the quality of paraphrase acquisition based on bilingual pivoting.

4.2. Qualitative Evaluation

Table 4 shows examples of paraphrases obtained for the word "creepy" (不気味な) using each method. It can be observed that for English, noisy paraphrases such as "kimonos", semantically unrelated to "creepy", were successfully removed from the top results by incorporating the neural method. Furthermore, for Japanese, noisy expressions such as "creepy 気味悪い" (creepy creepy) in the statistical method and "ような" (like/as if) in the neural method were effectively eliminated from the top rankings by combining GIZA++ and PMIAlign.

4.3. External Evaluation

In this section, we evaluate the usefulness of paraphrase databases by applying them to sentence pair modeling tasks. Following previous work (Sugiyama et al., 2025), we apply paraphrase-based contrastive learning, where paraphrases serve as positive examples and other sentences within the mini-batch as negative examples, to fine-tune pre-trained sentence encoders for sentence-pair modeling tasks.

4.3.1. Setting

Following previous work (Sugiyama et al., 2025), we evaluate the effectiveness of paraphrase-based contrastive learning across four tasks: product retrieval, sentence similarity estimation, recognizing textual entailment, and paraphrase identification (Tables 5 and 6). For pre-trained sentence encoders, as in previous work (Sugiyama et al., 2025),

English	Retrieval	Similarity		Entailment		Paraphrase	
	Shopping Queries	STS-B	SICK	SNLI	SICK	PAWS	Avg.
w/o Paraphrasing	0.654	0.824	0.815	**0.904**	0.858	0.913	0.828
PPDB 2.0	**0.655**	0.841	0.842	**0.904**	**0.866**	0.918	0.838
EhiMerPPDB[3]	**0.655**	**0.860**	0.829	**0.904**	0.858	**0.928**	0.839
Ours (PMIAlign)	**0.655**	0.859	**0.844**	0.900	0.860	0.925	**0.841**

Japanese	Retrieval	Similarity		Entailment		Paraphrase	
	Shopping Queries	JSTS	JSICK	JNLI	JSICK	PAWS-X	Avg.
w/o Paraphrasing	0.576	0.859	0.890	0.785	0.839	0.793	0.790
JPPDB[2]	0.586	0.857	0.896	0.824	0.854	0.795	0.802
EhiMerPPDB[3]	0.587	**0.861**	0.896	**0.828**	**0.856**	0.791	0.803
Ours (PMIAlign)	**0.588**	0.860	**0.901**	0.820	**0.856**	**0.803**	**0.805**

Table 7: External evaluation of paraphrase databases in sentence pair modeling tasks. The scores for "w/o Paraphrasing" and "PPDB 2.0" are taken from (Sugiyama et al., 2025).

we also used English BERT[14] (Devlin et al., 2019) and Japanese RoBERTa[15] (Liu et al., 2019).

Retrieval Product retrieval is a four-class classification task of the relationships between product titles and their search queries, and we employed both English and Japanese versions of the Shopping Queries dataset[16] (Reddy et al., 2022).

Similarity Sentence similarity estimation is a regression task that estimates the semantic similarity between two sentences, and we employed datasets of STS-B[17] (Cer et al., 2017) and SICK[18] (Marelli et al., 2014) for English and JSTS[19] (Kurihara et al., 2022) and JSICK[20] (Yanaka and Mineshima, 2022) for Japanese.

Entailment Recognizing textual entailment is a three-class classification task of semantic relationships between two sentences, and we employed datasets of SNLI[21] (Bowman et al., 2015) and SICK for English and JNLI[19] (Kurihara et al., 2022) and JSICK for Japanese.

Paraphrase Paraphrase identification is a two-class classification task of synonymity between two sentences, and we employed datasets of PAWS[22] (Zhang et al., 2019) for English and PAWS-X[22] (Yang et al., 2019) for Japanese.

For evaluation metrics, we employed the same metrics as previous work (Sugiyama et al., 2025): the micro-F1 score for the retrieval task, Spearman's rank correlation coefficient for the similarity task, and the macro-F1 score for both the entailment and paraphrase tasks. All other experimental details also followed (Sugiyama et al., 2025).

4.3.2. Result

Table 7 shows the experimental results. In both English and Japanese settings, paraphrase-based contrastive learning with our paraphrase database achieved the highest average performance. These experimental results suggest that we can acquire paraphrases that are effectively utilized to train better sentence encoders through contrastive learning.

5. Conclusion

In this study, we applied bilingual pivoting to a large-scale Japanese-English parallel corpus and constructed a Japanese paraphrase database containing 70 million pairs and an English paraphrase database containing 40 million pairs.[1] We successfully improved the quality of paraphrase acquisition by replacing the word alignment method for bilingual pivoting from a statistical approach to a neural approach. Although neural word alignment and traditional phrase extraction heuristics are incompatible, we achieved high-quality paraphrase acquisition by filtering and reranking noisy phrases.

[14] https://huggingface.co/google-bert/bert-base-uncased
[15] https://huggingface.co/rinna/japanese-roberta-base
[16] https://github.com/amazon-science/esci-data
[17] http://ixa2.si.ehu.eus/stswiki/stswiki/index.php/Special:Random.html
[18] https://zenodo.org/records/2787612
[19] https://github.com/yahoojapan/JGLUE
[20] https://github.com/verypluming/JSICK
[21] https://nlp.stanford.edu/projects/snli/
[22] https://github.com/google-research-datasets/paws

6. Bibliographical References

Fatemeh Azadi, Heshaam Faili, and Mohammad Javad Dousti. 2023. PMI-Align: Word Alignment With Point-Wise Mutual Information Without Requiring Parallel Training Data. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12366–12377.

Colin Bannard and Chris Callison-Burch. 2005. Paraphrasing with Bilingual Parallel Corpora. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 597–604.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A Large Annotated Corpus for Learning Natural Language Inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642.

Chris Callison-Burch, Philipp Koehn, and Miles Osborne. 2006. Improved Statistical Machine Translation Using Paraphrases. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 17–24.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation*, pages 1–14.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186.

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A. Smith. 2015. Retrofitting Word Vectors to Semantic Lexicons. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1606–1615.

Juri Ganitkevitch and Chris Callison-Burch. 2014. The Multilingual Paraphrase Database. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, pages 4276–4283.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The Paraphrase Database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 758–764.

Akio Hayakawa, Tomoyuki Kajiwara, Hiroki Ouchi, and Taro Watanabe. 2022. JADES: New Text Simplification Dataset in Japanese Targeted at Non-Native Speakers. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability*, pages 179–187.

Koki Horiguchi, Tomoyuki Kajiwara, Yuki Arase, and Takashi Ninomiya. 2024. Evaluation Dataset for Japanese Medical Text Simplification. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 219–225.

Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. 2020. SimAlign: High Quality Word Alignments Without Parallel Training Data Using Static and Contextualized Embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1627–1643.

Tomoyuki Kajiwara, Mamoru Komachi, and Daichi Mochihashi. 2017. MIPA: Mutual Information Based Paraphrase Acquisition via Bilingual Pivoting. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing*, pages 80–89.

Akihiro Katsuta and Kazuhide Yamamoto. 2018. Crowdsourced Corpus of Sentence Simplification with Core Vocabulary. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180.

Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. 2004. Applying Conditional Random Fields to Japanese Morphological Analysis. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 230–237.

Kentaro Kurihara, Daisuke Kawahara, and Tomohide Shibata. 2022. JGLUE: Japanese General Language Understanding Evaluation. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2957–2966.

Shuang Liu, Fang Liu, Clement Yu, and Weiyi Meng. 2004. An Effective Approach to Document Retrieval via Utilizing WordNet and Recognizing Phrases. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 266–272.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv:1907.11692*.

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A SICK Cure for the Evaluation of Compositional Distributional Semantic Models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, pages 216–223.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *Proceedings of the 1st International Conference on Learning Representations*.

Rina Miyata, Hyuga Koretaka, Hiroki Yamauchi, Daiki Yanamoto, Tomoyuki Kajiwara, Takashi Ninomiya, and Yasuhiro Nishiwaki. 2024. MATCHA: Parallel Corpus for Japanese Text Simplification Based on Professionally Simplified Articles. *Journal of Natural Language Processing*, 31(2):590–609.

Masahiro Mizukami, Graham Neubig, Sakriani Sakti, Tomoki Toda, and Satoshi Nakamura. 2014. Building a Free, General-domain Paraphrase Database for Japanese. In *Proceedings of the 17th Oriental Chapter of the International Committee for the Co-ordination and Standardization of Speech Databases and Assessment Techniques*, pages 1–4.

Makoto Morishita, Katsuki Chousa, Jun Suzuki, and Masaaki Nagata. 2022. JParaCrawl v3.0: A Large-scale English-Japanese Parallel Corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6704–6710.

Makoto Morishita, Jun Suzuki, and Masaaki Nagata. 2020. JParaCrawl: A Large Scale Web-Based English-Japanese Parallel Corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3603–3609.

Daiki Nishihara and Tomoyuki Kajiwara. 2020. Word Complexity Estimation for Japanese Lexical Simplification. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3114–3120.

Franz Josef Och and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51.

Ellie Pavlick and Chris Callison-Burch. 2016. Simple PPDB: A Paraphrase Database for Simplification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 143–148.

Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2015. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 425–430.

Jipeng Qiang, Yun Li, Yi Zhu, Yunhao Yuan, Yang Shi, and Xindong Wu. 2021. LSBert: Lexical Simplification Based on BERT. *Transactions on Audio, Speech, and Language Processing*, 29:3064–3076.

Chandan K. Reddy, Lluís Màrquez, Fran Valero, Nikhil Rao, Hugo Zaragoza, Sambaran Bandyopadhyay, Arnab Biswas, Anlu Xing, and Karthik Subbian. 2022. Shopping Queries Dataset: A Large-Scale ESCI Benchmark for Improving Product Search. *arXiv:2206.06588*.

Seiji Sugiyama, Risa Kondo, Tomoyuki Kajiwara, and Takashi Ninomiya. 2025. Paraphrase-based Contrastive Learning for Sentence Pair Modeling. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop)*, pages 400–407.

Renliang Sun, Wei Xu, and Xiaojun Wan. 2023. Teaching the Pre-trained Model to Generate Simple Texts for Text Simplification. In *Findings of the*

Association for Computational Linguistics: ACL 2023, pages 9345–9355.

Yui Suzuki, Tomoyuki Kajiwara, and Mamoru Komachi. 2017. Building a Non-Trivial Paraphrase Corpus Using Multiple Machine Translation Systems. In *Proceedings of ACL 2017, Student Research Workshop*, pages 36–42.

John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2016a. Charagram: Embedding Words and Sentences via Character n-grams. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1504–1515.

John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2016b. Towards Universal Paraphrastic Sentence Embeddings. In *Proceedings of the 4th International Conference on Learning Representations*.

Hitomi Yanaka and Koji Mineshima. 2022. Compositional Evaluation on Japanese Textual Entailment and Similarity. *Transactions of the Association for Computational Linguistics*, 10:1266–1284.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A Cross-lingual Adversarial Dataset for Paraphrase Identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3687–3692.

Tatsuya Zetsu, Tomoyuki Kajiwara, and Yuki Arase. 2022. Lexically Constrained Decoding with Edit Operation Prediction for Controllable Text Simplification. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability*, pages 147–153.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: Paraphrase Adversaries from Word Scrambling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1298–1308.

Using Syntax for the Semantic Representation of Sentences

Iskandar Boucharenc †[1], Eve Sauvage †[1,2], Thomas Gerald [1], Julien Tourille [2], Sabrina Campano [2], Cyril Grouin [1], Sophie Rosset [1]
[1]LISN, 507 rue du Belvédère, 91405 Orsay,
{name.surname}@lisn.fr
[2]EDF R&D, 7 Bd Gaspard Monge, 91120 Palaiseau
{name.surname}@edf.fr

Abstract

Deep learning methods in natural language processing often rely on statistical methods to tokenize texts before vectorization. This segmentation produces lexical subunits offering great flexibility. However, the reuse of identical tokens across words with different meanings can favor representations based on surface form rather than on linguistic information, especially semantics. This mismatch between semantics and surface form can lead to undesirable effects in language processing. To limit the influence of form on the semantics of vector representations, we propose an intermediate representation based on syntactic parsing that is more compact and more faithful to word meaning.

Keywords: Deep learning, Pre-Trained Models, Latent Representation, Constituent Analysis, Tokenization

1. Introduction

Deep learning methods for Natural Language Processing (NLP) rely on text vectorization to compute refined representations. The most commonly used method involves training statistical segmentation algorithms (tokenizers), such as SentencePiece (Kudo & Richardson, 2018), to break text into subunits that models can process. This tokenization results in subword-level segmentation, producing modular tokens that avoid errors caused by out-of-vocabulary words.

However, Tytgat et al. (2024) shows that deep models tend to favor the surface form of the lexicon over semantics in textual similarity computation. This preference for the surface form can lead to poor performance in tasks requiring fine-grained distinctions in sentence meaning, such as in Semantic Textual Similarity (STS) tasks or paraphrase detection (Muennighoff et al., 2023).

To address this issue, our hypothesis is to take advantage of structured linguistic data to make vector representation more faithful to the semantics of their text. We propose to improve the semantic information contained in lexical embeddings by using syntax which is closely related to semantics but benefits from larger parsing models. In this article, we explore three hypotheses: (i) the predominance of surface shape over semantics is linked to a lack of corelation between statistical segmentation and semantic information; (ii) Semantics-aware segmentation using constituency parsing could reduce the dependence of vector representations on surface form; finally, (iii) the merging of tokens representations belonging to the same constituent could help disambiguate vector representations. Exploring these hypotheses, we demonstrate that (a) models

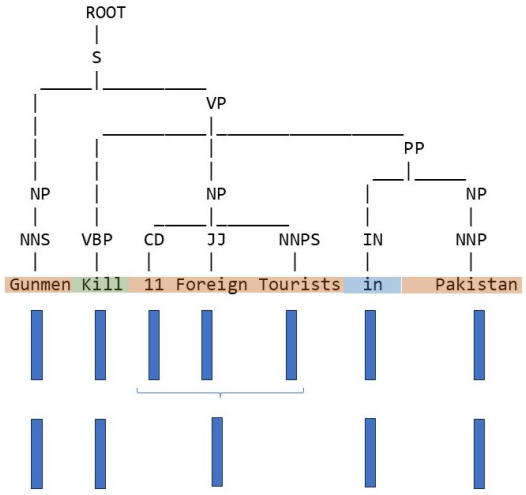


Figure 1: Example of token merging on a text from the *STSbenchmark* corpus. The different phrases are represented using color groups. The blue bars are embedded representation. Top blue bars represent the original token segmentation, while bottom blue bars represent the merged version.

are able to retain around 95% of their performances with deeply transformed representations; (b) Merging representations reduces the input sequence lengths from 26% to 43% in terms of number of tokens, depending on datasets; and (c) In the case of BERT, thanks to the length reduction due to merging token representations, a speed-up of 27.63% was achieved on average on the Semantic Textual Similarity (STS) subset of MTEB (Muennighoff et al., 2023).

The remainder of this article is organized as fol-

† These authors contributed equally

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 169–179

lows: section 2 presents the work related to our approach and our motivations. Section 3 details the methods used and the experimental protocol. Finally, we present our results and analyses in section 4, before concluding in section 5.

2. Related Works

In Natural Language Processing (NLP), methods derived from deep learning (DL) now dominate most tasks. These methods require representing text as vectors. The first vectorization approaches relied on one-hot encoding, a simple method that suffered from two major limitations: first, it produced very high-dimensional vectors; second, it did not account for the context in which words appeared.

With the rise of artificial neural networks, more efficient solutions have emerged, allowing to take context into vector representations through unsupervised learning (Mikolov et al., 2013; Pennington et al., 2014). These approaches exploit tasks such as masked language modeling (e.g., fill-in-the-blank text) to extract contextualized representations. However, while these methods improve context modeling, they still struggle to effectively capture the semantics of rare or out-of-vocabulary words. To overcome this limitation, Sennrich et al. (2016) proposed subword, or token, segmentation based on statistical rules, a technique for encoding words absent from the training vocabulary, albeit imperfectly.

However, this segmentation is based solely on statistical criteria, without linguistic grounding, which can introduce a bias in textual representation linked with surface form (Tytgat et al., 2024). Currently, the most efficient DL models are based on the Transformer architecture (Vaswani et al., 2017). Despite being highly parallelizable, these models exhibit quadratic complexity with respect to the length of the sequences processed. Therefore, subword segmentation poses an additional problem in terms of resource consumption, both computational and memory-wise (Dettmers et al., 2023).

Several recent studies suggest that the vocabulary size of a model follows a scaling law (Tao et al., 2024; Huang et al., 2025). Their experiments indicate that increasing vocabulary size could benefit the performance of models with a large number of parameters. Indeed, Tao et al. (2024) point out that the optimal vocabulary size is often underestimated, while Huang et al. (2025) demonstrates that vocabulary adaptation can be performed flexibly enough to decouple the input vocabulary from the output vocabulary. Even though merging token vectors does not change the vocabulary of the model, the model still has to learn to use new representations from its embedding layer.

Although the limitations of subunits are well known, the literature has mainly focused on model optimization (Dao et al., 2022; Dettmers et al., 2023). Nevertheless, some studies have explored input compression strategies to improve model efficiency. For example, in computer vision (CV), the visual token merging (TokenMerging (ToME)) approach (Bolya et al., 2023) has proven effective in speeding up processing. In NLP, Gee et al. (2023) studied the impact of increasing the vocabulary of language models by adding polylexical expressions, based on the frequency of n-gram occurrence. Our work follows this approach, but differs by the use of a non-statistical merging criterion based on linguistic information. Approaches based on a frequency criterion can, indeed, lead to performance losses on certain tasks (Kumar & Thawani, 2022).

3. Methodology

The aim of this study is to examine the impact of syntactic-based modifications of the token representations on encoder models. This requires adapting the model weights through a specific training process. This section details motivation for the use of syntactic parsing (3.1), the criteria and motivations guiding the representation modifications (3.2), the selected models and datasets (3.3), and the methodological choices adopted for model training (3.4).

3.1. Motivation

Syntax and semantics are two alternative views of one linguistic entity (namely its form and its meaning) with syntax providing the necessary means to assign meaning to utterances. In NLP tasks focusing on semantics, syntax is often used to determine boundaries. For example, Semantic Role Labeling (SRL), Named Entity Recognition (NER), Open Information Extraction (OIE) or Abstract Meaning Representation tasks can be enhanced by means of syntactic information (Cao & Clark, 2019; Dong et al., 2022; Huang et al., 2023). We based our study on these parallels and especially on the semantic aspect of constituents, also used in SRL.

Syntactic analysis has already been used with Large Language Models (Xu et al., 2021; Shen et al., 2021) to improve their understanding of language.

Reduction of the Impact of Surface Form The main motivation behind token merging lies in the predominant consideration of surface form in text encoding. A word or group of words belonging to a coherent semantic unit can be segmented into several tokens (see Figures 1 & 3), while words

with distinct meanings may share identical tokens. This artificial proximity in the representation space can lead to an overestimated similarity between semantically distant sentences. To address this problem, we propose an intermediate representation that mitigates the impact of surface form by merging tokens belonging to the same semantic unit (cf. figure 1).

With this method, we only increase the vocabulary size artificially by adding an intermediate representation after the embedding phase. Thus, by integrating a merging step into the model, we seek to evaluate its impact on performance and input data compression.

Input Compression Merging tokens in a text reduces the number of tokens to be processed by the model and, consequently, reduces the length of the sequences. Since Transformer models have quadratic complexity with respect to sequence length, this reduction represents a significant computational advantage.

3.2. Modifiying Token Representations

To test our hypotheses, we propose an approach to merge the representations of certain tokens based on syntactic analysis. This section presents the method used, the criteria selected, and the implementation details.

Merging Criteria Given the importance of semantic representation in language models and the semantic lens adopted by Mikolov et al. (2013) or Peters et al. (2018) to explain lexical embeddings, it seems appropriate to adopt a merging criterion based on semantics. A division into lexemes, the smallest units of meaning, might seem like a logical approach. However, there are neither reference corpora nor readily available lexeme segmentation models. Moreover, such an approach would not allow the consideration of MultiWord Expressions (MWE) and would require a profound modification of the representations used by the models.

We therefore propose an approach based on constituent analysis rather than phrase parsing for its greater versatility. This analysis allows us to extract phrases from different sentences on different levels, which are often considered independent semantic units as well as syntactic units. They are identifiable using constituent tests that highlight their structural autonomy (topicalization, substitution, dislocation, splitting, etc.). Constituent analysis makes it possible to identify and reconstruct independent semantic units within a sentence, providing a clearer representation of its internal structure by incorporating additional information that is not captured by dependency analysis alone. Indeed, dependency analysis lacks information on membership in grammatical categories, as theorized by Chomsky (1956). Unlike an approach limited solely to named entities, constituent segmentation offers broader coverage across the entire text. Although this method can identify a large number of embeddings to merge, it results in a substantial modification of the distribution of vectors, which could be detrimental to the performance of the model.

We choose to base the merger on the noun (NP), prepositional (PP), and verbal (VP) phrases closest to the leaves of the syntax tree. This selection allows us to obtain constituents of significant size, thus significantly reducing the length of the models' input sequences. For the sake of simplification, we exclude adverbial and adjectival phrases, even though they are among the five generally identified types of phrases. Indeed, these phrases are often included in other phrases, allowing us to consider them without processing them separately.

Implementation Details For constituent analysis, models are provided by Stanza (Qi et al., 2020), one of the few analyzers available for this task. Once constituents are identified, their boundaries are aligned with the output of the tokenizer. The obtained alignment is used to build merging matrices that send the original representation to the new one. The matrix is composed of 1 for unchanged representation, $\frac{1}{t}$ for constituents formed by t tokens to merge, 0 elsewhere. The remainder of the process is performed using matrix operations from the PyTorch library, enabling fast batched merging.

3.3. Tasks, Datasets and Models

Tasks The relevance of the obtained representations is evaluated along three criteria. First, training and evaluating models on the STSBenchmark dataset, from the Massive Textual Embedding Benchmark (MTEB) (Muennighoff et al., 2023), with and without merged representations to evaluate if the new representations are learnable. Second, after training, models and representations are evaluated on the test sets of the other STS tasks from MTEB to assess their robustness and generalizability. Third, an evaluation is conducted following the work of Tytgat et al. (2024) to gauge whether the surface form is less predominant after merging. The main purpose of this evaluation in our experimental setting is to verify that the representation of a synonym for a term is closer to that of the term in question than that of a near form (homophone, homograph or cognate), which we call paronyms. We can argue that using paronyms instead of homographs to evaluate the impact of similar tokens' apparition on sentence similarity is under effective.

We use the corpus proposed by Tytgat et al. (2024), which we will refer to as the SYNPAR corpus. We conduct our experiments in English. All datasets are presented in Table 1.

Training Datasets The models are trained during two phases, a pretraining phase on a subset of Wikipedia and a finetuning phase on STSBenchmark.

1. Pre-training on a subset uniformly sampled from Wikipedia EN[1] due to the richness and diversity of the topics covered, making this corpus a suitable basis for initial model training.
2. Fine-tuning on MTEB/STSBenchmark. We only use the train split for the finetuning in order to avoid data contamination.

Models We use the models studied by Tytgat et al. (2024), namely BERT (Devlin et al., 2019)[2] and all-miniLM-L6-v2[3]. The choice of these models is based on several criteria:

- their lightweight nature, which facilitates the adaptation and reproducibility of experiments
- the availability and transparency of their training data
- their prior training on limited amounts of data, thus limiting the impact of acquired biases

While `all-miniLM-L6-v2` is adapted to text similarity tasks, `bert-base-uncased` offers an optimal basis for experiments because of its task agnostic pretraining.

3.4. Model Training

To allow the models to adapt to the merged representation, we modify their weights. Our approach is based on two steps: (i) pre-training based on a task analogous to MLM (*Masked Language Modeling*) and (ii) fine-tuning on a sentence similarity task, as summarized in table 3.

Pre-training Merging representations automatically results in a change in the length distribution of representation vectors (see Figure 2), reducing the length of sequences by 42.83% on average in the Wikipedia corpus. To mitigate the impact of this change, a common approach (applied, for example, during a change of domain) is to continue the pre-training of the model. Given that the models used were initially pre-trained on an MLM task, we adapt this task to our representation in the new context.

This task adaptation is necessary because the merged representations no longer directly correspond to vocabulary tokens, making the traditional objective of the pre-training task using cross-entropy on tokens inapplicable. We therefore opt for a continuous approach to this task (*i.e.* the token vector representation is masked instead of its index). The objective of this pre-training is to allow the model to acquire a semantic understanding of the merged token groups. Rather than optimizing cross-entropy between the prediction probabilities of masked tokens, we train the model to reconstruct the vector resulting from merging before passing it into the network: the representations are first merged, then the resulting groups are masked to be reconstructed by the model. The representation produced by the final layer is processed by a perceptron, then compared to the corresponding input vector. The L^2 distance between the output of the perceptron for a masked representation and the merged representation is retained as the loss function. This is summarized below.

$$\mathcal{L}(X, \tilde{X}) = \mathrm{MSE}(X, \mathcal{F}(X))$$

with $\mathcal{F} = p \circ \mathcal{M}$ where p is a perceptron whose input and output dimensions are the dimensions of the model $\mathcal{M}$, $\tilde{X} = \mathcal{F}(X)$. The model and perceptron are trained on a randomly selected subset of Wikipedia representing 14M tokens. We save the model producing the best average loss on 3200 texts.

Adaptation In parallel with pre-training the models to understand merged tokens, we propose adapting these models for the sentence similarity task by incorporating the merging step. Unlike the masked token task, the model's loss can be calculated similarly to the training of the Siamese networks proposed by Reimers & Gurevych (2019). More precisely, we evaluate the cosine similarity between the sentence representations of a pair of sentences after applying the merging procedure. Then, the MSE (Mean Squared Error) loss is calculated between the ground-truth similarity and the similarity predicted by our approach. We use the Spearman correlation (r_s) on the validation set as a stopping criterion.

3.5. Evaluation

We use the SynPar dataset to evaluate the semantic expressiveness of our representations. Indeed, this dataset emphasises the contrast between lexical forms and meanings, providing a good founda-

[1] https://huggingface.co/datasets/legacy-datasets/wikipedia

[2] https://huggingface.co/google-bert/bert-base-uncased

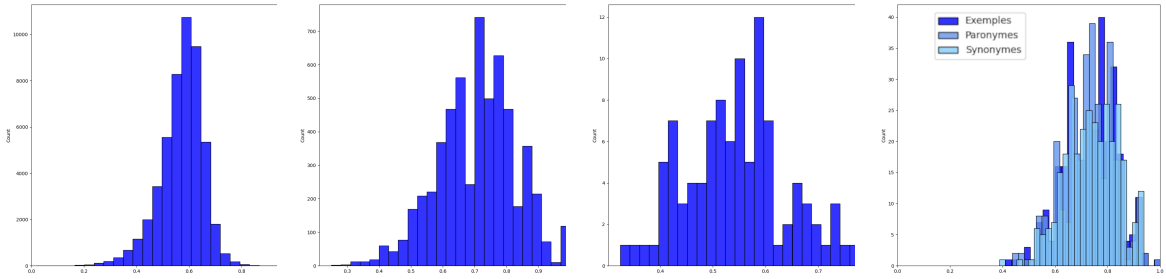
[3] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

Dataset	Description	# Test Examples
SynPar	Sentence sets in which one of the words is derived into four variants: original, synonym, paronym (cf. table 2)	372
STSBench.	Compilation of various pair of sentences for Semantic Similarity	5.75k
BIOSSES	Semantic sentence similarity estimation system for the biomedical domain	100
SICKR	Sentences Involving Compositional Knowldedge	9.93k
STS12	Semantic similarity from existing paraphrase datasets and machine translation evaluation resources	2.23k
STS13	Typed Semantic Similarity	1.5k
STS14	Emotion labelled corpus of tweets	3.75k
STS15	Semantic Similarity with Referential Translation Machines	3k
STS16	Sentiment Analysis in Twitter	1.19k
SummEval	Evaluation of machine generated summaries	100

Table 1: Summary of the different tasks evaluated. Futher information can be found in the MTEBenchmark paper (Muennighoff et al., 2023)

original	paronym	synonym
Please **accept** this gift as a sign of my gratitude.	Please **except** this gift as a sign of my gratitude.	Please **receive** this gift as a sign of my gratitude.

Table 2: SynPar dataset exemple



(a) Wikipedia (b) STSBenchmark (c) BIOSSES (d) SynPar

Figure 2: Size of the merged texts compared to the original texts. The size ratio is read on the x-axis (percentage), and the number of sequences with this ratio is read on the y-axis.

Model	conf.	merged	pre-trained	finetuned
name	zs	✗ / ✓	✗	✗
	pt	✓	✓	✗
	ft	✗ / ✓	✗	✓
	pt-ft	✓	✓	✓

Table 3: Summary of the experiments carried out. Pre-training (pt) is done on 50,000 Wikipedia texts, and fine-tuning (ft) is done on the STSBenchmark game. The abbreviation zs stands for zero-shot evaluation. We will use these terms in the rest of the article for clarity.

tion for semantic evaluation. We verify the learnability of merged representations with an ablation study on STSBenchmark. In this study, we compare pretraining, fine-tuning, and zero-shot settings. The robustness of the performances is evaluated on the other MTEB STS datasets.

ABX Test In order to compare our results with those of Tytgat et al. (2024), we use the ABX test described by Carlin et al. (2011) and Schartz et al. (2013). This test measures, between 0 and 100, the similarity between an element A and an element B having minimal differences with it, in comparison with an element X, which has major differences (equation 1).

$$100 \times \frac{\mathrm{Card}(\cos(A,B) \geq \cos(A,X))}{\#\mathrm{Examples}} \quad (1)$$

Because we need to obtain one vector per sentences, we have to agregate the intermediate representations obtained. We experiment with three

different agregations configuration : using the sentence's mean representation, using the pooler of the model representations or using only the CLS of the sentence.

Spearman Correlation As in the evaluation of the MTEB Benchmark, we use the Spearman (1904) correlation r_s for the evaluation on this corpus. We exclude the Pearson correlation, following the recommendations of Reimers et al. (2016).

4. Results & Analyses

4.1. Results on STSBenchmark

To evaluate the results on the training corpus, as in the literature, we calculated the Spearman correlation for the different models whose results are reported in table 4 using the notations defined in table 3. The table is divided into two parts, one for each base model. The first two rows represent the model without constituent merging. The second row corresponds to a fine-tuning without merging as a control. The following three rows can be interpreted as an ablation study by successively adding fine-tuning (ft) and pre-training (pt-ft) phases.

		merge	r_s validation	r_s test	Δ (test)
bert	zs	✗	59.32	47.29	–
	ft	✗	**87.48**	**84.80**	**+37.60**
	zs	✓	48.46	40.13	-7.16
	ft	✓	83.96	79.05	+31.76
	pt-ft	✓	84.21	79.65	+32.36
miniLM	zs	✗	86.72	82.03	–
	ft	✗	**89.51**	**85.49**	**+3.46**
	zs	✓	59.24	42.81	-39.22
	ft	✓	85.05	78.10	-3.93
	pt-ft	✓	86.08	81.03	-1.0

Table 4: Spearman correlation (r_s) for the different models and performance difference with the zero-shot evaluation (Δ).Results are grayed out when the inference representation does not match the training representation of the models (original model on merged representation or vice versa). The best scores are in bold, and the second bests are underlined. *cf.* table 3 for the explanation of the model names.

Maintaining the zero-shot framework is no longer feasible due to changes in the representation distribution, as evidenced by the significant drops in performance for the zs evaluation on merged representations. In the case of the BERT model, merging yields good results, even when fine-tuning the model without merging. The increase in the correlation metric with the compression ratio of 29.91% validates our assumptions for the BERT model. However, in the case of MiniLM, any merging attempt results in a drop in performance. Although pre-training puts this drop into perspective and the average compression ratio remains attractive, it is more difficult to assess this model. Note that the objectives of the two models differ slightly. The all-miniLM-L6-v2 version is designed to generate efficient general-purpose embeddings and has a more complex pretraining procedure, which could explain the difficulty in recovering from the drop in performance. BERT, on the other hand, is intended to be fine-tuned on downstream tasks.

4.2. Results on Other MTEB Tasks

Further evaluations are conducted on other MTEB tasks. Firstly, Table 5 compares results on the remaining STS tasks of MTEB. All models are fine-tuned on the STSBenchmark training set and evaluated on their respective test sets. Average score and inference times are reported in the last two columns. Both models experience a slight drop in performance but retain 94.45% (BERT) and 96.34% (MiniLM) of the unmerge counterpart. The major drop in performance is due to BIOSSES. Indeed, the analysis of relative sequence lengths shows that, due to the presence of scientific terms, the text is overtokenized, as shown in Figure 3. The same effects, but to a lesser extent, occur with STS16, which draws examples from Twitter. Regarding inference time, while BERT shows a 27.63% speed-up on the STS subset of MTEB on average, MiniLM shows a 19.56% slowdown, with $0.0088\,\mathrm{s}$ more per example on average. Secondly, Table 6 compares the results obtained by BERT and MiniLM on the Summeval dataset. Another difference between the two models is that Fine-Tuning BERT with token merging on STSBenchmark improves its results, while MiniLM suffers from this tuning. Regarding inference time, BERT still benefits from a speed-up, as does MiniLM this time.

4.3. Results on the SynPar Corpus

For SynPar dataset, we find results similar to those in the previous section on table 7: the gains with BERT are much greater. While MiniLM shows a decrease in ABX scores, BERT outperforms MiniLM when average of the vectors is taken as a representation (see BERT-PT-FT in the table 7). The 2.6-point improvement in score over the zero-shot version confirms that merging representations is beneficial for the BERT model, both for compressing sequences and for avoiding confusion due to paronymy. On MiniLM, an improvement is visible on unmerged data after pretraining the model on merging for `cls` configuration. There is, therefore, room for improvement. Nonetheless, it shows the

	merge	STSB	BIO.	SIC.	STS13	STS14	STS15	STS16	STS12	Avg.	Ratio	t.(sec)
bert	✗	**84.60**	**72.73**	**73.76**	**87.59**	**87.78**	86.95	**81.03**	**78.26**	**81.58**	–	0.154
	✓	79.65	57.73	72.02	84.83	85.67	**87.06**	75.82	73.83	77.05	94.45	**0,111**
miniLM	✗	**85.69**	**83.15**	**77.82**	**88.25**	85.98	**89.68**	**82.85**	**75.53**	**83.61**	–	**0.045**
	✓	81.03	66.06	73.67	86.21	**86.68**	88.59	79.95	73.83	79.53	95.12	0.054

Table 5: Spearmanr results on the STS datasets. Models are finetuned on STSBenchmark and then evaluated on the remaining data. The average processing times are reported in the last column. BIOSSES and SICKR are abbreviated BIO. and SIC. respectively

		merge	r_s	t.(sec)
bert	zs	✗	29.82	–
		✓	15.37	2.11
	ft	✗	27.79	2.40
		✓	**32.49**	**1.29**
miniLM	zs	✗	**30.81**	–
		✓	25.77	**1.10**
	ft	✗	25.33	1.14
		✓	23.13	1.27

Table 6: Results on SumEval dataset comparing with (✗) and without (✓) merging tokens and, the zero-shot (zs) approach versus the fine-tuned (ft) configuration on STSBenchmark.

		merge	configurations		
			mean	pooler	cls
bert	zs	✗	66,88	56,17	66,56
		✓	60,39	55,84	58,44
	pt	✗	62,01	64,61	65,26
		✓	61,04	57,80	57,80
	pt-ft	✗	67,53	63,64	**67,21**
		✓	**69,48**	**65,26**	66,23
miniLM	zs	✗	**67,53**	**68,83**	68,83
		✓	59,74	64,29	63,31
	pt	✗	65,91	67,53	**69,16**
		✓	58,17	62,99	62,99
	pt-ft	✗	64,29	65,91	66,88
		✓	58,77	62,66	64,94

Table 7: Results of ABX tests on the SynPar corpus comparing the addition of token merging. Results are grayed out when the inference representation does not match the training representation of the models (original model on fused representation or vice versa). The best scores are in bold.

particular nature of MiniLM embeddings and their over-optimisation.

Two interesting points emerge from these experiments : when only the embedding layer is considered, merging representations improves ABX scores from 52.6% to 56.82% for `bert-base-uncased` and from 49.02% to 50.32% for `miniLM`. The second point is that the overall best result for `MiniLM` happens for PT without merged representation on CLS. This result is surprising because of the inadequacy between the pre-trained representation and the model inference representation. The best BERT CLS ABX score also occurs on inadequate data. It could mean that model learning merged representation may help even for unmerged representations on CLS, which is only indirectly affected by the merge.

4.4. Discussion

Based on the observation that language models calculate higher similarity with paronyms than with synonyms, we proposed a method to merge tokens based on linguistic criteria. The evaluation took three criteria into account: the first, performance on a common NLP task, STSBenchmark. The second, robustness and generalization to unseen data. The third, more qualitative, focused on the distinction between paronyms and synonyms in the ABX test. In both cases, the method proved conclusive after a pre-training and fine-tuning phase on the STS task. Considering the evaluation on the STS-Benchmark dataset, the pre-trained and fine-tuned BERT model yields the best results with a compression ratio of 29.91%. MiniLM shows a 1-point drop in performance but retains the compression property. In both cases, the experiments demonstrated fairly good stability to changes in learning rate.

In contrast to our initial assumption (the improvement of sentence similarity performance with merged intermediate representations), the qualitative evaluation was conclusive only for BERT. Indeed, these results show that the STS task helps distinguish paronyms in this model, as fine-tuning improves performance. In contrast, MiniLM shows a decrease in the ABX score, except for the pt-ft version evaluated without merging. This result may be symptomatic of a poor training method. We assume that improvements could be achieved by pretraining on more data, which currently contains only 14 million tokens. Note that our results on STS cannot be considered an improvement in the task since, unlike the original models, we are moving away from the zero-shot framework. Thus, merging representations achieves results that retain 94% of the score of the original models while reducing the length of the input sequences significantly.

Original: "T47D, MCF-7, Skbr3, HeLa, and Caco-2 cells were transfected by electroporation as described previously."

Tokenized: ‘t’, ‘##47’, ‘##d’, ‘,’, ‘mc’, ‘##f’, ‘-’, ‘7’, ‘,’, ‘sk’, ‘##br’, ‘##3’, ‘,’, ‘he’, ‘##la’, ‘,’, ‘and’, ‘ca’, ‘##co’, ‘-’, ‘2’, ‘cells’, ‘were’, ‘trans’, ‘##fect’, ‘##ed’, ‘by’, ‘electro’, ‘##por’, ‘##ation’, ‘as’, ‘described’, ‘previously’, ‘.’

Merged: ‘t47d’,‘,’, ‘mcf-7’,‘,’, ‘skbr3’,‘,’, ‘hela’, ‘and caco-2 cells’, ‘were transfected’, ‘by electroportation’, ‘as described previously’

Figure 3: Original, tokenized, and merged processing of a BIOSSES Benchmark example.

Merging tokens based on a constituent analysis criterion allows, across the four datasets, an average reduction of 42.83%, 29.91%, 45.66%, and 26.42%, respectively, in the size of Wikipedia, STS-Benchmark, BIOSSES, and SynPar. More precise distribution visualization can be observed in figure 2, which shows the histograms of the relative sizes after merging. As explained in section 3, this reduction therefore allows for longer contexts to be taken as input without increasing the complexity of the models.

Inference Acceleration

As suspected, the merging procedure generally lacks generalizability. This drawback is salient on the domain-specific BIOSSES dataset. However, reducing sequence lengths automatically consequently reduces inference time. On average, an increase in inference speed of 27.63% is observed with BERT. The highest increase in speed is obtained on the BIOSSES dataset, which shows a significant reduction in sequence lengths (see Figure 2d). It should be noted, however, that this comes with a preprocessing time that can be costly, since the constituent parser is quite slow. Preprocessing time ranges from $0.11\,\mathrm{s}$ to $0.24\,\mathrm{s}$. Possible solutions will be discussed in the section 5.

Domain Specific Generalization

The performance ratio rises from 94.45% to 96.36% for BERT and from 95.15% to 97.35% for MiniLM when we exclude the domain-specific BIOSSES dataset, which comprises biomedical data. However, it should be noted that the model has not seen any domain-specific data other than that included in the general Wikipedia subset and the STS-Benchmark dataset. Furthermore, the measured speed-up of the forward pass is three times faster. This is because the BERT Tokenizer tends to overdivide scientific words and concepts. The Figure 3 shows an overtokenization of domain-specific language inputs such as Biomedical terms. Although the drop in performance is significant, our observations regarding the benefits of adaptation lead us to hypothesize that the merging procedure with an adaptation phase could be highly beneficial for specialized vocabulary.

5. Conclusions

To improve the semantic representation of texts, we proposed a method for merging vector representations. Based on the assumption that constituent analysis provides semantically relevant information, we merge the token representations of the constituents. This method yields interesting results with BERT and mixed results with MiniLM. In both cases, sequence compression is significant, suggesting the need for further experiments.

We must acknowledge certain limitations to our approach. Token merging is only applied to neighboring tokens in the input sequence. It is difficult to consider the merging of two disjoint tokens within the framework of language models. However, even if the constituent analysis methods selected here do not account for them, some constituents are discontinuous (Coavoux, 2020). Selecting a linguistic criterion for token merging results in heavy text preprocessing, which reduces the performance gains due to the reduced sequence size. Using faster parsers could greatly improve our method.

The experiments conducted here consider segmentation based on noun, verb, and prepositional phrases. Other types of segmentation may also be of interest and require testing. Indeed, constituent parsing is primarily motivated by English and lacks literature in other languages. The proposed method should be language-agnostic, as it relies solely on merging representations based on external linguistic criteria. However, the experiments were conducted only on English corpora and could benefit from being extended to other languages.

The results obtained with MiniLM raise questions about our pre-training procedure and generalization of performances across models. A corpus containing more tokens or more similar to the model’s pre-training corpus would be interesting. The nature of the corpora can influence the results obtained. It would therefore be relevant to construct a corpus similar to the SynPar corpus from the *STSBenchmark* data to ensure corpus similarity.

To ensure the generalizability of this method across models and architectures, further experiments should be conducted.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

Acknowledgments

This work was granted access to the HPC resources of IDRIS under the allocation 2025-AD011014970R1 made by GENCI.).

6. Bibliographical References

Bolya, D., Fu, C.-Y., Dai, X., Zhang, P., Feichtenhofer, C., and Hoffman, J. Token merging: Your vit but faster. *ICLR*, abs/2210.09461, 2023. URL https://openreview.net/forum?id=JroZRaRw7Eu.

Cao, K. and Clark, S. Factorising AMR generation through syntax. In Burstein, J., Doran, C., and Solorio, T. (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2157–2163, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1223. URL https://aclanthology.org/N19-1223/.

Carlin, M., Thomas, S., Jansen, A., and Hermansky, H. Rapid evaluation of speech representations for spoken term discovery. In *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

Chomsky, N. Three models for the description of language. *IRE Transactions on Information Theory*, 2(3):113–124, 1956. doi: 10.1109/TIT.1956.1056813.

Coavoux, M. Qu'apporte BERT à l'analyse syntaxique en constituants discontinus ? une suite de tests pour évaluer les prédictions de structures syntaxiques discontinues en anglais (what does BERT contribute to discontinuous constituency parsing ? a test suite to evaluate discontinuous constituency structure predictions in English). In Benzitoun, C., Braud, C., Huber, L., Langlois, D., Ouni, S., Pogodalla, S., and Schneider, S. (eds.), *Actes de la 6e conférence conjointe Journées d'Études sur la Parole (JEP, 33e édition), Traitement Automatique des Langues Naturelles (TALN, 27e édition), Rencontre des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RÉCITAL, 22e édition). Volume 2 : Traitement Automatique des Langues Naturelles*, pp. 189–196, Nancy, France, 6 2020. ATALA et AFCP. URL https://aclanthology.org/2020.jeptalnrecital-taln.17/.

Dao, T., Fu, D. Y., Ermon, S., Rudra, A., and Ré, C. Flashattention: Fast and memory-efficient exact attention with io-awareness. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/67d57c32e20fd0a7a302cb81d36e40d5-Abstract-Conference.html.

Dettmers, T., Pagnoni, A., Holtzman, A., and Zettlemoyer, L. QLoRA: Efficient finetuning of quantized LLMs. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=OUIFPHEgJU.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Dong, K., Sun, A., Kim, J.-J., and Li, X. Syntactic multi-view learning for open information extraction. In Goldberg, Y., Kozareva, Z., and Zhang, Y. (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 4072–4083, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.272. URL https://aclanthology.org/2022.emnlp-main.272/.

Gee, L., Rigutini, L., Ernandes, M., and Zugarini, A. Multi-word tokenization for sequence compression. In Wang, M. and Zitouni, I. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 612–621, Singapore, December

2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-industry.58. URL https://aclanthology.org/2023.emnlp-industry.58/.

Huang, H., Zhu, D., Wu, B., Zeng, Y., Wang, Y., Min, Q., and Zhou, X. Over-tokenized transformer: Vocabulary is generally worth scaling. *CoRR*, abs/2501.16975, 2025. doi: 10.48550/ARXIV.2501.16975. URL https://doi.org/10.48550/arXiv.2501.16975.

Huang, P., Zhao, X., Hu, M., Tan, Z., and Xiao, W. T 2 -NER: A two-stage span-based framework for unified named entity recognition with templates. *Transactions of the Association for Computational Linguistics*, 11:1265–1282, 2023. doi: 10.1162/tacl_a_00602. URL https://aclanthology.org/2023.tacl-1.72/.

Kudo, T. and Richardson, J. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In Blanco, E. and Lu, W. (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 66–71, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-2012. URL https://aclanthology.org/D18-2012/.

Kumar, D. and Thawani, A. BPE beyond word boundary: How NOT to use multi word expressions in neural machine translation. In Tafreshi, S., Sedoc, J., Rogers, A., Drozd, A., Rumshisky, A., and Akula, A. (eds.), *Proceedings of the Third Workshop on Insights from Negative Results in NLP*, pp. 172–179, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.insights-1.24. URL https://aclanthology.org/2022.insights-1.24/.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. Efficient estimation of word representations in vector space. In Bengio, Y. and LeCun, Y. (eds.), *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013. URL http://arxiv.org/abs/1301.3781.

Muennighoff, N., Tazi, N., Magne, L., and Reimers, N. MTEB: Massive text embedding benchmark. In Vlachos, A. and Augenstein, I. (eds.), *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 2014–2037, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.eacl-main.148. URL https://aclanthology.org/2023.eacl-main.148/.

Pennington, J., Socher, R., and Manning, C. D. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014. URL http://www.aclweb.org/anthology/D14-1162.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. Deep contextualized word representations. In Walker, M., Ji, H., and Stent, A. (eds.), *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1202. URL https://aclanthology.org/N18-1202/.

Qi, P., Zhang, Y., Zhang, Y., Bolton, J., and Manning, C. D. Stanza: A python natural language processing toolkit for many human languages. In Celikyilmaz, A. and Wen, T.-H. (eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pp. 101–108, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-demos.14. URL https://aclanthology.org/2020.acl-demos.14/.

Reimers, N. and Gurevych, I. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Inui, K., Jiang, J., Ng, V., and Wan, X. (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410.

Reimers, N., Beyer, P., and Gurevych, I. Task-oriented intrinsic evaluation of semantic textual similarity. In Matsumoto, Y. and Prasad, R. (eds.), *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 87–96, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee. URL https://aclanthology.org/C16-1009/.

Schartz, T., Peddinti, V., Bach, F., Jansen, A., Hermansky, H., and Dupoux, E. Evaluating speech features with the minimal-pair abx task : Analysis of the classical mfc/plp pipeline. In *INTERSPEECH 2013 : 14th Annual Conference of the*

International Speech Communication Association, 2013.

Sennrich, R., Haddow, B., and Birch, A. Neural machine translation of rare words with subword units. In Erk, K. and Smith, N. A. (eds.), *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1715–1725, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1162. URL https://aclanthology.org/P16-1162/.

Shen, Y., Tan, S., Sordoni, A., Reddy, S., and Courville, A. Explicitly modeling syntax in language models with incremental parsing and a dynamic oracle. In Toutanova, K., Rumshisky, A., Zettlemoyer, L., Hakkani-Tur, D., Beltagy, I., Bethard, S., Cotterell, R., Chakraborty, T., and Zhou, Y. (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1660–1672, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.132. URL https://aclanthology.org/2021.naacl-main.132/.

Spearman, C. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101, 1904. ISSN 00029556. URL http://www.jstor.org/stable/1412159.

Tao, C., Liu, Q., Dou, L., Muennighoff, N., Wan, Z., Luo, P., Lin, M., and Wong, N. Scaling laws with vocabulary: Larger models deserve larger vocabularies. In Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., and Zhang, C. (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 114147–114179. Curran Associates, Inc., 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/cf5a019ae9c11b4be88213ce3f85d85c-Paper-Conference.pdf.

Tytgat, J., Wisniewski, G., and Betrancourt, A. Évaluation de la similarité textuelle : Entre sémantique et surface dans les représentations neuronales. In Balaguer, M., Bendahman, N., Ho-dac, L.-M., Mauclair, J., G Moreno, J., and Pinquier, J. (eds.), *Actes de la 31ème Conférence sur le Traitement Automatique des Langues Naturelles, volume 1 : articles longs et prises de position*, pp. 85–96, Toulouse, France, 7 2024. ATALA and AFPC. URL https://aclanthology.org/2024.jeptalnrecital-taln.6.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Xu, Z., Guo, D., Tang, D., Su, Q., Shou, L., Gong, M., Zhong, W., Quan, X., Jiang, D., and Duan, N. Syntax-enhanced pre-trained model. In Zong, C., Xia, F., Li, W., and Navigli, R. (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 5412–5422, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.420. URL https://aclanthology.org/2021.acl-long.420/.

Modeling Word-Internal Structures: Morphological Segmentation Across 58 Languages

Vojtěch John, Benjamin Reeves, Zdeněk Žabokrtský
Charles University, Faculty of Mathematics and Physics, Institute of Formal and Applied Linguistics
Malostranské náměstí 25, Prague, Czech Republic
benjamin.reeves286@student.cuni.cz
{john, zabokrtsky}@ufal.mff.cuni.cz

Abstract

We present the largest multilingual experiment to date on word-to-morph segmentation, covering 58 typologically diverse languages. We describe a newly compiled collection of linguistically annotated resources for the task, providing broad coverage and enabling systematic cross-lingual evaluation. Second, we train two neural models on surface morphological segmentation, achieving 81% average word accuracy on the original datasets, slightly outperforming previous methods. Experiments on custom test sets reveal substantial variation in performance, highlighting the need for further harmonization and more robust multilingual approaches.

Keywords: morph, morphology, morphological segmentation, multilingual NLP

1. Introduction

Morph segmentation provides a practical intermediate layer between raw word forms and higher-level linguistic annotation. By dividing words into smaller meaning-bearing units, it can help reduce data sparsity, especially in morphologically rich languages (which can improve e.g. machine translation in low-resource settings (Mager et al., 2022)), and make patterns of inflection and derivation more transparent in structured datasets (as seen in e.g. (McCarthy et al., 2020)). Typically, morphological structure implicitly affects how lexical resources are organized, how annotations are aligned across tools and languages, and how systems are evaluated. Making word-internal structure explicit and cross-linguistically analyzable can therefore support clearer annotation guidelines, more controlled comparisons between approaches, and also provide valuable insights into language evolution, including phenomena arising from language contact. In some settings, morphological segmentation has proved to improve performance on downstream task, especially for low-resource languages.

Previous research, with the SIGMORPHON 2022 Shared Task on Morpheme Segmentation (Batsuren et al., 2022) being one of the most ambitious collective efforts, has still covered only a limited range of typologically diverse languages, which restricts the generalizability of its findings. Studying word-internal morphological segmentation across many languages is therefore important, as morphological systems differ substantially in structure and transparency. Truly polysynthetic or strongly introflexive languages remain largely beyond the current empirical scope and represent a longer-term goal, but even broadening coverage toward more agglutinative and fusional languages already should facilitate more robust evaluation, clearer identification of language-specific biases, and more broadly applicable modeling assumptions.

Although a multilingual perspective on segmentation is essential, advancement in the field is constrained by the fragmented and inconsistently annotated landscape of available morphological data. Data resources existing for individual languages vary widely in format, annotation guidelines, size, and quality, making cross-linguistic comparisons difficult and inconsistent. To enable robust evaluation, reproducible research, and meaningful benchmarking, there is a pressing need for standardized, harmonized, and high-quality morphologically segmented datasets that can be applied consistently across languages and systems. Our goal is to bring the world of morphological segmentation closer to the level of standardization and interoperability already achieved in the dependency syntax community through Universal Dependencies (de Marneffe et al., 2021).

Our long-term research questions focus on how to represent word-internal structure consistently across typologically diverse languages and how to standardize datasets and annotation practices to enable robust cross-linguistic evaluation, as well as how to develop systems that can perform this segmentation automatically. Looking further ahead, we aim to connect morph segmentation with other linguistic layers, explore typological effects on model performance, and establish benchmarks and metrics that support reproducible and comparable results.

The remainder of the paper is organized as follows. Section 2 summarized related work. Sec-

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 180–190

tion 3 describes the data collection that we use in our experiments. Section 4 details the evaluation setup. Section 5 presents the modeling approaches and experimental results, which are further interpreted in Section 6. Section 7 concludes and presents future work.

2. Basic notions and Related work

As noted by Haspelmath (2020), the notion of *morpheme* is subject to considerable terminological ambiguity in linguistics. In this work, we adopt the following distinction. A *morpheme* is an abstract linguistic unit defined as the smallest unit of language that carries semantic or grammatical function. A *morph*, by contrast, is the concrete realization of a morpheme in actual language use, i.e., a specific phonological or orthographic form that instantiates a given morpheme.

Morphs may constitute entire words (e.g. *house*, consisting of a single root morph), or they may form parts of more complex words (e.g. *un+kind+ness*, which contains three morphs). The central element of a word is the *root morphs*, conveying its core lexical meaning. Additional morphs, if present, are classified relative to the root. A root may be preceded by one or more *prefixes* (e.g. *un-* in *un+kind*), and followed by one or more *suffixes* (e.g. *-ness* in *kind+ness*). A suffix expressing inflectional categories such as number or tense (e.g. *-ed* in *walk+ed*) is often referred to as an *ending*. In compounds containing multiple roots, linking elements or *interfixes* may appear between the roots (e.g. *speed+o+meter*).

The term *morph* has been used in linguistics for less than a century. Although early attempts at a rigorous delimitation of the morpheme date back at least to the 1940s (Hockett, 1947), the notions of morph and morpheme are still frequently employed only implicitly and informally in linguistic practice. Nevertheless, there exist numerous attempts at lexicographic description of the morphemic structure of words in individual languages over the past decades, though these efforts remain largely isolated. Somewhat surprisingly, even though morphology is often considered a lower level of abstraction compared to syntax, there is still no *de facto* standard for morphological segmentation comparable, in terms of both community size and coverage of languages, to the Universal Dependencies project in syntax (de Marneffe et al., 2021). A few multilingual harmonization initiatives have emerged in the last decade, such as the UniMorph collection (McCarthy et al., 2020), the UniSegments collection (Žabokrtský et al., 2022), and the SIGMORPHON 2022 Shared Task on Morpheme Segmentation (Batsuren et al., 2022), but these resources remain considerably smaller, providing complete morphological segmentation for samples of words for 14, 32, and 9 languages, respectively.

Approaches to automatic morphological segmentation closely mirror the broader development of NLP. Focusing first on methods aimed at producing linguistically interpretable divisions of words into smaller units, the earliest approaches were rule-based (Klenk and Langer, 1989), followed by supervised machine learning methods (Na, 2015), and more recently by deep learning sequence labeling techniques (Girrbach, 2022) and large language models (Pranjić et al., 2024). In parallel, unsupervised segmentation has been an important research line, with Morfessor (Creutz and Lagus, 2005) as its most widely known representative; however, the resulting segmentations are typically of limited linguistic interpretability.

With the rise of deep learning as the dominant paradigm in NLP, the need to segment words into shorter units emerged rapidly. This was driven not by linguistic concerns, but by a practical technical constraint: the input layer of neural networks cannot feasibly accommodate the full vocabulary of a language. These shorter units, referred to as *subwords* (Sennrich et al., 2016), do not correspond to morphs or morphemes, yet subword segmentation methods are frequently used as baselines in morphologically informed NLP research.

Evaluating morphological segmentation is challenging due to the inherently ambiguous nature of morphemes and the variety of possible segmentation schemes. Commonly used metrics include precision, recall, and F1-score at the morph boundary level, which measure how well predicted boundaries match a gold-standard segmentation (Creutz and Lagus, 2005; Narasimhan et al., 2015). Another common approach is word-level accuracy, which requires the entire word to be segmented correctly, providing a stricter measure of performance (Batsuren et al., 2022). In multilingual settings, evaluation is often complicated by the heterogeneity of annotation schemes across languages, differing tokenization conventions, and the presence of allomorphy or non-concatenative morphology. Fully capturing linguistic validity remains an open challenge.

3. Data Collection

This section describes a newly compiled collection of diverse language resources relevant to morphological segmentation. The collection makes use of the harmonization annotation scheme of UniSegments 1.0 (Žabokrtský et al., 2022), and reuses some of its data components too (after a revision). Compared to UniSegments 1.0, the expanded collection covers nearly twice as many languages. A subset of the current collection – comprising only

converted resources whose original licenses permit modification and redistribution – is now publicly available as UniSegments 1.5 (John et al., 2026).

Table 1 summarizes all integrated resources (including those that we cannot distribute further) along with their main qualitative and quantitative properties. The original datasets differ along multiple dimensions, some of which are described in the following paragraphs.

The collected resources reflect the landscape of modern morphological research, bridging the gap between small-scale, linguistically grounded "gold" data and large-scale, automatically generated "silver" and "bronze" datasets. This collection encompasses scripts as diverse as Malayalam, Greek, and Kanji, with a primary objective of maximizing the volume of fully segmented data. Within this framework, resources providing segmentation for roots, inflections, and derivations are categorized as Complete. Conversely, those where the stem remains unsegmented are designated as Partial. These partial resources typically feature a single segmentation boundary per word, usually distinguishing between the stem and its inflectional or derivational affixes. While the granularity of segment labeling varies significantly across the collection, most labeled resources identify at least the prefix, suffix, and root, though a substantial portion remains unlabeled.

Data quality is largely a function of the segmentation methodology. Manual segmentation serves as the gold standard, whereas automatic methods, while more scalable, frequently introduce noise. To mitigate this, many resources in this set were either automatically generated and human-verified or developed through rule-based methods to ensure cross-linguistic consistency. Notably, 55 resources contain manual annotations, 15 of which originate from Metamorphosis, a newly developed manually annotated dataset. These expert-led approaches generally yield higher-fidelity data than unsupervised or semi-supervised machine learning models. Consequently, the collection as a whole represents a strategic trade-off between high-volume noisy data and high-quality, small-scale datasets, with resource sizes ranging from 200 to over 740,000 wordforms or lemmas.

The final corpus comprises 85 resources covering 58 languages from 14 language families, including Niger-Congo, Indo-European, Uralic, Turkic, and Austronesian. While the set spans four continents, there is a recognized geographic imbalance: North and South American languages are absent, and African and Asian languages are underrepresented relative to European ones. Furthermore, while 20 distinct scripts are represented, five resources have been transliterated into the Latin alphabet, a process that may obscure native orthographic cues essential for morphological analysis.

This diverse linguistic composition enables a granular analysis of morphological systems across the typological spectrum. This ranges from the strictly isolating profile of Mandarin Chinese to the extreme polysynthetic complexity of Adyghe. The dataset further represents fusional structures through Latin, Czech, and Greek, alongside the highly productive agglutinating systems of Turkish, Hungarian, and Korean. Beyond broad typological classification, the collection captures specialized morphological processes: Amharic provides a case study for non-concatenative (root-and-pattern) morphology, while Indonesian, Tagalog, Swahili, Zulu, Xhosa, and Hindi illustrate diverse reduplicative patterns. Furthermore, extensive compounding is evident in the Germanic and Uralic families, as well as Chinese, while the inclusion of the Romance group and Modern Greek facilitates the study of cliticization.

4. Evaluation strategies

Although superficially harmonized, the resources differ in several important respects. Firstly, some of the datasets cover only a highly specific part of lexicon (e.g. *SlavickovaDict* or *CroDeriV* covers only verb infinitives). Hence, it is unclear how models trained on such data would behave on a more representative sample. In the resources, that were not originally developed for morphological segmentation (e.g. *DerIvaTario* for word formation or *Uniparser* for morphological analysis), the quality of the segmentation varies both in and across the resources. For example, in DerIvaTario, the infinitives are usually not segmented, because they are often regarded as the unmotivated words. Furthermore, for some of the languages, we have only resources with partial segmentation. Most of the resources are segmented wordlists, often without any further annotation. As a result, we did not attempt to resolve differences in segmentation between homonyms. Since we aim for broad coverage, we have used all of the resources irrespective of their overall quality. In this section, we describe issues this causes for evaluation of the automatic segmenters, evaluation strategies we have adopted to mitigate them, and the limitations this imposes on the interpretation of the results.

As a result of the widely differing quality of the resources, evaluating the automatic segmentation on a test set sampled from the same resource as the train set is problematic. The metrics can devolve to mere consistency of the automatic segmentation with the resource, rather than actual quality of segmentation. It is unclear how well such results approximate the performance on complete segmentation of arbitrary words. For exam-

ISO	Family	Resource Name	Script	Count	Meth.	C/P	Cat.	Avg M/W	Avg M.Len	Avg W.Len	Reference
ady	NWC	MorphAGram	Cyrillic	999	A+H	C	–	5.92	2.08	12.30	(Eskander et al., 2020)
aka	NC	LDC_RLP	Latin	2047	A+R	C	–	2.38	2.28	5.44	(Tracey and Strassel, 2020)
amh	AA	Amharic	Latin^T	135184	M	C	R	3.75	2.45	9.21	(Yeshambel et al., 2021)
bel	IE	Slounik	Cyrillic	35219	M	C	D, I, R, X	3.77	2.43	9.17	(Morozov et al., 2025)
ben	IE	kcis	Bengali	822	M+A	P	R, S	2.00	2.81	5.61	(LTRC, 2018)
cat	IE	Morphynet	Latin	19777	M+A	P	P, R, S	1.90	4.92	9.32	(Batsuren et al., 2021)
ces	IE	SlavickovaDict	Latin	14582	M+A	C	–	2.59	2.24	5.79	(Slavíčková, 1975)
ces	IE	Morphynet	Latin	351251	M+A	P	P, R, S	2.17	4.43	9.61	(Batsuren et al., 2021)
deu	IE	MorphoChallenge	Latin	2863	M+A	C	–	2.98	3.51	10.46	(Kurimo et al., 2010)
deu	IE	CELEX	Latin	47583	M	C	Int, P, R, S	2.37	2.72	6.44	(Baayen et al., 1995)
deu	IE	Morphynet	Latin	236544	M+A	P	P, R, S	2.10	5.30	11.16	(Batsuren et al., 2021)
ell	IE	GreekAnnot.Dict.	Greek	8419	A+S	C	D, I, R	2.40	3.12	7.49	(UI and NTUA, 2021)
eng	IE	MorphoChallenge	Latin	3355	M+A	C	–	2.30	3.67	8.44	(Kurimo et al., 2010)
eng	IE	CELEX	Latin	43957	M	C	Int, P, R, S	1.40	3.87	5.42	(Baayen et al., 1995)
eng	IE	MorphoLex	Latin	68549	M	C	P, R, S	2.21	3.77	8.34	(Sánchez-Gutiérrez et al., 2018)
eng	IE	Morphynet	Latin	387418	M+A	P	P, R, S	2.18	4.86	10.58	(Batsuren et al., 2021)
fas	IE	PerSegLex	Perso-Ar.	45369	M	C	–	2.12	3.20	6.80	(Ansari et al., 2019)
fin	Ur	MorphoChallenge	Latin	3859	M+A	C	–	3.43	3.81	13.04	(Kurimo et al., 2010)
fin	Ur	Morphynet	Latin	740711	M+A	P	P, R, S	2.31	5.65	13.07	(Batsuren et al., 2021)
fra	IE	MorphoLex	Latin	15954	M+A	C	P, R, S	2.03	3.15	6.40	(Mailhot et al., 2020)
fra	IE	demonette	Latin	81494	M+A	P	C, Int, P, R, S	1.72	4.70	8.20	(Hathout and Namer, 2014)
fra	IE	Morphynet	Latin	132094	M+A	P	P, R, S	1.84	5.66	10.42	(Batsuren et al., 2021)
hbs	IE	Morphynet	Latin, Cyrillic	4915	M+A	P	P, R, S	1.84	4.60	8.46	(Batsuren et al., 2021)
hin	IE	kcis	Devanagari	1603	M+A	P	R, S	1.89	3.60	6.78	(LTRC, 2018)
hin	IE	LDC_RLP	Latin^T	2029	A+H	C	R	1.72	2.33	4.40	(Tracey and Strassel, 2020)
hrv	IE	CroDeriV	Latin	15657	M	C	E, Int, P, R, S	4.08	2.30	9.66	(Šojat et al., 2014)
hun	Ur	LDC_RLP	Latin	2027	A+H	C	R	2.48	3.42	8.47	(Tracey and Strassel, 2020)
hun	Ur	Morphynet	Latin	28176	M+A	P	P, R, S	2.23	4.13	9.21	(Batsuren et al., 2021)
hye	IE	Uniparser	Armenian	593542	A+R	P	P, R, S	2.56	3.75	9.59	(Arkhangelskiy et al., 2012)
ind	AN	LDC_RLP	Latin	2035	A+H	C	–	1.79	4.23	7.58	(Tracey and Strassel, 2020)
ita	IE	DerIvaTario	Latin	11287	M	C	R	2.80	3.86	10.92	(Talamo et al., 2016)
ita	IE	Morphynet	Latin	80532	M+A	P	P, R, S	2.32	2.65	6.14	(Batsuren et al., 2021)
jpn	Ja	MorphAGram	Kanji, Hirigana	999	M	C	–	2.24	1.43	3.21	(Eskander et al., 2020)
kan	Dr	kcis	Kannada	25953	M+A	P	Int, P, R, S	4.36	2.46	9.45	(LTRC, 2018)
kat	Ka	MorphAGram	Mkhedruli	999	A+H	C	–	3.04	2.38	7.24	(Eskander et al., 2020)
kor	Ko	LDC2004T03	Latin^T	6455	A+H	C	–	1.77	5.47	9.66	(Han, 2004)
kpv	Ur	Uniparser	Cyrillic	216128	A+R	P	P, R, S	2.52	3.38	8.54	(Arkhangelskiy et al., 2012)
lat	IE	WFL	Latin	27998	M+A	P	P, R, S	2.39	3.53	8.48	(Litta Modignani et al., 2023)
mal	Dr	kcis	Malayalam	33657	M+A	P	P, R, S	1.99	4.70	12.40	(LTRC, 2018)
mar	IE	kcis	Devanagari	32523	M+A	P	Int, R, S	2.54	3.27	8.30	(LTRC, 2018)
mdf	Ur	Uniparser	Cyrillic	104672	A+R	P	R, S	2.28	3.91	8.90	(Arkhangelskiy et al., 2012)
mhr	Ur	Uniparser	Cyrillic	251373	A+R	P	P, R, S	2.44	3.52	8.59	(Arkhangelskiy et al., 2012)
mon	Mo	Morphynet	Cyrillic	11428	M+A	P	R, S	1.93	4.40	8.48	(Batsuren et al., 2021)
myv	Ur	Uniparser	Cyrillic	162564	A+R	P	R, S	2.40	3.76	9.03	(Arkhangelskiy et al., 2012)
nbl	NC	Sadilar	Latin	14753	A+H	C	–	3.70	2.63	9.72	(Gaustad and McKellar, 2024)
nld	IE	CELEX	Latin	100620	M+A	C	Int, P, R, S	2.44	4.34	10.81	(Baayen et al., 1995)
nso	NC	Sadilar	Latin	5338	A+H	C	–	2.52	3.04	7.68	(Gaustad and McKellar, 2024)
pol	IE	Morphynet	Latin	58711	M+A	P	P, R, S	1.48	6.66	9.83	(Batsuren et al., 2021)
por	IE	Morphynet	Latin	11774	M+A	P	P, R, S	1.49	7.18	10.67	(Batsuren et al., 2021)
rus	IE	Crosslexica	Cyrillic	27873	M	C	D, I, R	3.76	2.57	9.67	(Bolshakov, 2013)
rus	IE	KuznetsEfremDict	Cyrillic	73447	M	C	Int, P, R, S	4.34	2.27	9.85	(Kuznetsova and Efremova, 1986)
rus	IE	Morphynet	Cyrillic	93039	M+A	P	P, R, S	3.86	2.67	10.28	(Batsuren et al., 2021)
rus	IE	Tikhonov	Cyrillic	96046	M	C	D, I, R, X	2.87	2.11	6.05	(Alimardanova, 2024)
sot	NC	Sadilar	Latin	5667	A+H	C	–	2.46	3.00	7.38	(Gaustad and McKellar, 2024)
spa	IE	MATS	Latin	3541	A	C	–	2.13	3.07	6.55	(García et al., 2025)
spa	IE	Morphynet	Latin	214483	M+A	P	P, R, S	2.11	4.95	10.45	(Batsuren et al., 2021)
ssw	NC	Sadilar	Latin	14493	A+H	C	–	3.57	2.74	9.79	(Gaustad and McKellar, 2024)
swa	NC	LDC_RLP	Latin	2023	A+H	C	R	3.27	2.44	7.98	(Tracey and Strassel, 2020)
swe	IE	Morphynet	Latin	87024	M+A	P	P, R, S	2.52	3.89	9.79	(Batsuren et al., 2021)
tam	Dr	LDC_RLP	Latin^T	2029	A+H	C	–	1.99	4.92	9.82	(Tracey and Strassel, 2020)
tgk	IE	Uniparser	Cyrillic	230646	A+R	P	P, R, S	2.08	3.71	7.71	(Arkhangelskiy et al., 2012)
tgl	AN	LDC_RLP	Latin	2005	A+H	C	–	2.17	3.54	7.69	(Tracey and Strassel, 2020)
tsn	NC	Sadilar	Latin	5946	A+H	C	–	2.48	3.18	7.90	(Gaustad and McKellar, 2024)
tso	NC	Sadilar	Latin	5202	A+H	C	–	2.08	3.79	7.87	(Gaustad and McKellar, 2024)
tur	Tu	MorphoChallenge	Latin^T	6643	M+A	C	–	3.45	3.05	10.51	(Kurimo et al., 2010)
udm	Ur	Uniparser	Cyrillic	401382	A+R	P	Inf, R, S	2.59	3.47	8.97	(Arkhangelskiy et al., 2012)
uig	Tu	thuuymorph	Uyghur Ar.	20958	A+R	C	–	2.17	4.22	9.16	(Gulinigeer et al., 2021)
ven	NC	Sadilar	Latin	5273	A+H	C	–	1.93	3.84	7.42	(Gaustad and McKellar, 2024)
xho	NC	Sadilar	Latin	15480	A+H	C	–	3.96	2.35	9.32	(Gaustad and McKellar, 2024)
zul	NC	Sadilar	Latin	14798	A+H	C	–	3.81	2.45	9.33	(Gaustad and McKellar, 2024)

Table 1: Resources used in the training of the models. Languages codes specified by ISO 639-3 are used in the ISO column. Abbreviations used in the Family column: NWC = Northwestern Caucasian, NC = Niger-Congo, AA = Afro-Asiatic, IE = Indo-European, Ur = Uralic, AN = Austronesian, Ja = Japonic, Dr = Dravidian, Ka = Kartvelian, Ko = Koreanic, Mo = Mongolic, ST = Sino-Tibetan, Tu = Turkic. ^{T} in the Script means transcription from the original writing system to the Latin script. Count = the number of segmented lemmas or wordforms available in the resource. Abbreviations used in the Meth. column: A = Automatic, Un = Unsupervised, R = Rule-Based, M = Manual, H = Human Reviewed. Abbreviations used in the C/P column: C = Complete (fully segmented), P = Partial (only some boundaries). Abbreviations for distinguished morph types in the Cat. column: R = Root, D = Derivational, I = Inflectional, X = Other, S = Suffix, P = Prefix, C = Connector, E = Ending, Int = Interfix, Inf = Infix. Avg. M/W = average number of morphs per lemma or word form. Avg M.Len = the average number of characters per morph. Avg W.Len = the average number of characters per lemma or word form.

ISO	Family	Script	Avg M/W	Avg M.Len	Avg W.Len
ces	IE	Latin	4.05	2.3	9.31
deu	IE	Latin	2.5	4.01	10.01
ell	IE	Greek	2.24	3.5	7.84
eng	IE	Latin	1.93	4.46	8.62
epo	Con.	Latin	2.41	2.59	6.26
fra	IE	Latin	1.66	5.86	9.75
hrv	IE	Latin	2.64	2.12	5.61
ita	IE	Latin	1.58	7.12	11.28
lat	IE	Latin	2.5	2.77	6.94
pol	IE	Latin	2.71	2.23	6.05
rus	IE	Cyrillic	1.65	6.32	10.45
spa	IE	Latin	2.25	3.48	7.81
tel	Dr	Telugu	1.49	4.82	7.16
ukr	IE	Cyrillic	2.64	2.15	5.67
zho	ST	Hanzi	1.93	1.05	2.12

Table 2: Metamorphosis resources. All resources are manually annotated, complete, and contain 200 wordforms. Languages identified by ISO 639-3 code. Avg M/W = average number of morphs per lemma or word form, Avg M.Len = the average number of characters per morph., Avg W.Len = the average number of characters per lemma or word form.

ple, we expect nominally good results for automatically generated datasets, (the model might reverse-engineer the original, often simple pipeline), and for datasets with partial segmentation. The comparability across resources is also limited.

However, the evaluation using manually segmented data, even if they are consistent across languages, is not necessarily more accurate. It might be strongly influenced by differences in annotation approaches between training and test data. Take as an illustration two pairs of highly accurate datasets: *MorphoLex* with *MorphoChallenge* for English, and *Tikhonov* with *KuznetsEfremDict* for Russian. From the 1941 words common to the training subsets of *eng-MorphoLex* and *eng-MorphoChallenge*, the same segmentation appears for only 72.7%. In the case of *Tikhonov* and *KuznetsEfremDict*, it is even less (namely 58.3%, out of 53,048 words). For example, the English word *stabilized* is segmented as *stabil+iz+ed* in *MorphoChallenge* but as *stabil+ize+d* in *MorphoLex*.

We have decided to evaluate our models in two ways. Firstly, we have split the original resources into two mutually exclusive subsets – training data and test data. The test data from the resources were sampled randomly according to the size of the resource (2000 words for resources with more than 5000 words, else 200 words), while the rest of the resource was used for training. Secondly, we have evaluated our models on a small, manually annotated dataset, where available (see Table 2). As evaluation metrics, we use whole-word accuracy, and precision, recall and F1 on boundaries between morphs.

5. Experiments and Results

Although transformers-based models have been used for segmentation in a high-resource setting (Batsuren et al., 2022), the performance gap between transformer-based models and simpler LSTM-based models appears to be quite small. Among other advantages of LSTM-based models is the comparatively low number of parameters, and therefore faster training time. Such an architecture also seems more appropriate for small-size datasets, where it achieved competitive results (Olbrich and Žabokrtský, 2025). We have thus opted for employing two LSTM-based architectures, trained on each resource separately.

Similarly to Girrbach (2022), we encode segmentation as character classification, using the BMES encoding (Ruokolainen et al., 2014). Each character is classified as beginning (B), middle (M), or end (E) of morpheme, or as single-character morpheme (S). For example, the word *be-ing-s* would be annotated as *b:B, e:E, i:B, n:M, g:E, s:S*.

The first architecture we have used is a two-layer LSTM, as presented by Girrbach (2022) and implemented in the publicly available Python package *morphseg* (Winkelman et al., 2026). Models using this architecture (TüSeg) have achieved competitive results in the SIGMORPHON 2022 Shared Task on Morpheme Segmentation (Batsuren et al., 2022). The implementation we use automatically removes non-alphabetical characters. We automatically restore most of these characters before evaluation. As we don't get any prediction for these characters, we treat them by default as morpheme boundaries. While not the best performing architecture in the original shared task, it might be more appropriate for our setting, which includes small datasets and concentrates of surface segmentation.

We have also created our custom architecture (CNN-LSTM), inspired by Olbrich and Žabokrtský (2025). Our models, after embedding the individual characters (with embedding size 128), apply convolutions with kernel sizes 1 to 8 and 64 channels each in order to capture local dependencies. Concatenated outputs of these convolutions are then fed into a bidirectional LSTM layer (of dimension 512, close to the optimum according to (Olbrich and Žabokrtský, 2025)), followed by a dense layer and a final dense classification layer. We train for 100 epochs with batch size 64 for large resources, 16 for small resources, with early stopping on word accuracy on validation set. It isn't strictly guaranteed that the output of the neural network will be a valid BMES-encoded segmentation. There are multiple possible strategies of extracting a valid segmentation from the output of the neural network, including conditional random fields or Viterbi decod-

	Morfessor				CNN-LSTM				TüSeg			
resource	WAcc	MBPrec	MBRec	MBF	WAcc	MBPrec	MBRec	MBF	WAcc	MBPrec	MBRec	MBF
ady-MorphAGram	0.10	0.76	0.64	0.69	**0.61**	0.96	0.95	**0.95**	0.58	0.94	0.96	**0.95**
aka-LDC_RLP	0.24	0.47	0.85	0.60	**0.80**	0.91	0.91	**0.91**	0.74	0.87	0.89	0.88
amh-Amharic	0.04	0.49	0.37	0.42	**0.76**	0.93	0.95	**0.94**	0.75	0.93	0.95	**0.94**
bel-Slounik	0.05	0.50	0.43	0.46	0.91	0.98	0.98	**0.98**	**0.92**	0.98	0.98	**0.98**
ben-kcis	0.14	0.32	0.74	0.44	**0.86**	0.87	0.87	**0.87**	0.74	0.87	0.76	0.81
cat-Morphynet	0.07	0.15	0.69	0.24	**0.87**	0.91	0.93	**0.92**	**0.87**	0.91	0.91	0.91
ces-Morphynet	0.24	0.32	0.67	0.44	**0.96**	0.97	0.98	**0.98**	0.93	0.95	0.96	0.96
ces-SlavickovaDict	0.00	0.59	0.44	0.51	0.94	0.99	0.99	**0.99**	**0.95**	0.99	0.99	**0.99**
deu-CELEX	0.15	0.35	0.72	0.47	**0.88**	0.95	0.96	**0.95**	0.83	0.91	0.94	0.93
deu-MorphoChallenge	0.01	0.27	0.80	0.41	**0.59**	0.83	0.86	**0.84**	0.53	0.84	0.81	0.82
deu-Morphynet	0.09	0.22	0.56	0.31	**0.93**	0.95	0.97	**0.96**	0.69	0.78	0.81	0.80
ell-GreekAnnotatedDict	0.09	0.26	0.71	0.39	**0.83**	0.92	0.92	**0.92**	0.60	0.86	0.76	0.80
eng-CELEX	0.12	0.20	0.71	0.32	0.86	0.92	0.92	**0.92**	**0.87**	0.91	0.94	**0.92**
eng-MorphoChallenge	0.07	0.22	0.88	0.36	**0.62**	0.81	0.79	**0.80**	0.59	0.79	0.80	0.79
eng-MorphoLex	0.17	0.31	0.76	0.44	0.92	0.96	0.97	0.96	**0.93**	0.96	0.97	**0.97**
eng-Morphynet	0.19	0.31	0.74	0.44	**0.90**	0.95	0.96	**0.95**	**0.90**	0.94	0.97	**0.95**
fas-PerSegLex	0.11	0.25	0.74	0.38	**0.85**	0.93	0.91	**0.92**	**0.85**	0.92	0.92	**0.92**
fin-MorphoChallenge	0.01	0.24	0.60	0.34	0.58	0.86	0.86	**0.86**	**0.59**	0.86	0.86	**0.86**
fin-Morphynet	0.05	0.24	0.40	0.30	**0.99**	1.00	1.00	**1.00**	0.97	0.98	1.00	0.99
fra-demonette	0.03	0.07	0.39	0.12	**0.89**	0.95	0.90	**0.92**	**0.89**	0.95	0.90	**0.92**
fra-MorphoLex	0.03	0.14	0.72	0.23	0.66	0.74	0.74	0.74	**0.67**	0.77	0.73	**0.75**
fra-Morphynet	0.08	0.17	0.75	0.28	**0.86**	0.92	0.91	**0.91**	**0.86**	0.89	0.92	0.90
hbs-Morphynet	0.07	0.18	0.84	0.30	**0.85**	0.89	0.90	**0.89**	0.74	0.83	0.80	0.81
hin-kcis	0.18	0.44	0.86	0.58	0.82	0.91	0.85	0.88	**0.83**	0.91	0.88	**0.89**
hin-LDC_RLP	0.04	0.19	0.89	0.32	**0.82**	0.88	0.88	**0.88**	0.75	0.90	0.77	0.83
hrv-CroDeriV	0.00	0.39	0.28	0.33	**0.93**	0.98	0.99	**0.98**	**0.93**	0.98	0.99	**0.98**
hun-LDC_RLP	0.03	0.23	0.88	0.36	**0.58**	0.80	0.77	**0.78**	0.54	0.79	0.71	0.75
hun-Morphynet	0.18	0.33	0.76	0.46	**0.89**	0.93	0.95	0.94	**0.89**	0.94	0.95	**0.95**
hye-Uniparser	0.18	0.47	0.61	0.53	**0.84**	0.95	0.94	**0.94**	0.83	0.91	0.97	**0.94**
ind-LDC_RLP	0.04	0.15	0.88	0.26	**0.89**	0.93	0.90	**0.91**	0.83	0.88	0.89	0.88
ita-DerIvaTario	0.05	0.26	0.53	0.35	**0.70**	0.90	0.89	**0.89**	0.67	0.89	0.87	0.88
ita-Morphynet	0.03	0.06	0.47	0.11	0.91	0.88	0.95	**0.92**	**0.92**	0.92	0.93	**0.92**
jpn-MorphAGram	0.41	0.60	0.69	0.65	0.67	0.76	0.92	0.83	**0.72**	0.81	0.91	**0.86**
kan-kcis	0.12	0.63	0.41	0.50	0.72	0.91	0.94	0.92	**0.74**	0.92	0.94	**0.93**
kat-MorphAGram	0.04	0.34	0.69	0.46	0.58	0.88	0.83	0.85	**0.59**	0.87	0.87	**0.87**
kor-LDC2004T03	0.05	0.12	0.74	0.21	**0.82**	0.86	0.84	**0.85**	0.66	0.64	0.78	0.70
kpv-Uniparser	0.18	0.44	0.61	0.51	**0.81**	0.91	0.94	**0.92**	0.80	0.90	0.94	**0.92**
lat-WFL	0.08	0.30	0.56	0.39	**0.71**	0.84	0.91	**0.88**	0.70	0.89	0.85	0.87
mal-kcis	0.04	0.31	0.42	0.36	**0.62**	0.86	0.83	**0.84**	0.60	0.83	0.82	0.83
mar-kcis	0.19	0.44	0.71	0.54	0.82	0.91	0.93	**0.92**	**0.83**	0.91	0.94	**0.92**
mdf-Uniparser	0.14	0.34	0.65	0.44	**0.89**	0.94	0.95	**0.95**	0.88	0.93	0.96	0.94
mhr-Uniparser	0.09	0.35	0.59	0.44	**0.71**	0.86	0.86	**0.86**	**0.71**	0.88	0.82	0.85
mon-Morphynet	0.33	0.32	0.92	0.48	**0.97**	0.98	0.99	**0.99**	**0.97**	0.98	0.99	**0.99**
myv-Uniparser	0.10	0.34	0.59	0.43	**0.85**	0.91	0.95	**0.93**	0.83	0.94	0.91	0.92
nbl-Sadilar	0.07	0.49	0.51	0.50	**0.64**	0.90	0.91	**0.91**	0.62	0.88	0.93	0.90
nld-CELEX	0.27	0.42	0.69	0.52	**0.93**	0.97	0.98	**0.97**	0.92	0.97	0.97	**0.97**
nso-Sadilar	0.04	0.20	0.56	0.30	**0.84**	0.92	0.93	**0.92**	0.82	0.90	0.93	**0.92**
pol-Morphynet	0.03	0.08	0.70	0.14	0.85	0.82	0.85	**0.84**	**0.86**	0.84	0.84	**0.84**
por-Morphynet	0.03	0.07	0.67	0.12	0.85	0.84	0.86	**0.85**	**0.86**	0.87	0.83	**0.85**
rus-Crosslexica	0.08	0.65	0.48	0.55	**0.96**	0.99	0.99	**0.99**	**0.96**	0.99	0.99	**0.99**
rus-KuznetsEfremDict	0.06	0.68	0.43	0.53	**0.89**	0.98	0.98	**0.98**	0.88	0.98	0.98	**0.98**
rus-Morphynet	0.03	0.09	0.55	0.15	0.86	0.89	0.87	0.88	**0.87**	0.89	0.89	**0.89**
rus-Tikhonov	0.05	0.55	0.43	0.48	**0.91**	0.98	0.98	**0.98**	0.89	0.98	0.97	0.97
sot-Sadilar	0.04	0.21	0.62	0.32	**0.82**	0.92	0.92	**0.92**	**0.82**	0.91	0.92	**0.92**
spa-MATS	0.01	0.22	0.82	0.35	**0.67**	0.81	0.84	**0.83**	0.66	0.81	0.84	0.82
spa-Morphynet	0.07	0.22	0.57	0.31	**0.95**	0.97	0.97	**0.97**	**0.95**	0.97	0.98	**0.97**
ssw-Sadilar	0.04	0.45	0.54	0.49	**0.77**	0.94	0.96	**0.95**	0.75	0.93	0.95	0.94
swa-LDC_RLP	0.01	0.35	0.59	0.44	**0.66**	0.91	0.90	**0.91**	**0.66**	0.91	0.91	**0.91**
swe-Morphynet	0.10	0.41	0.66	0.51	**0.94**	0.97	0.98	**0.98**	**0.94**	0.97	0.98	**0.98**
tam-LDC_RLP	0.02	0.15	0.72	0.25	**0.64**	0.73	0.75	**0.74**	0.58	0.75	0.65	0.70
tgk-Uniparser	0.24	0.44	0.73	0.55	**0.88**	0.92	0.98	**0.95**	0.87	0.92	0.98	**0.95**
tgl-LDC_RLP	0.01	0.18	0.88	0.30	**0.72**	0.86	0.79	**0.83**	0.68	0.81	0.80	0.81
tsn-Sadilar	0.04	0.20	0.58	0.30	**0.78**	0.87	0.91	**0.89**	0.76	0.89	0.86	0.88
tso-Sadilar	0.02	0.16	0.68	0.25	**0.78**	0.88	0.87	**0.88**	0.77	0.87	0.88	0.87
tur-MorphoChallenge	0.02	0.31	0.54	0.40	**0.44**	0.83	0.81	**0.82**	0.38	0.79	0.76	0.78
udm-Uniparser	0.14	0.40	0.59	0.48	**0.89**	0.97	0.93	**0.95**	**0.89**	0.94	0.97	**0.95**
uig-thuuymorph	0.12	0.27	0.64	0.38	**0.92**	0.96	0.97	**0.96**	**0.92**	0.96	0.97	**0.96**
ven-Sadilar	0.04	0.15	0.62	0.24	**0.74**	0.82	0.84	**0.83**	0.73	0.80	0.85	0.82
xho-Sadilar	0.10	0.54	0.55	0.55	**0.80**	0.95	0.95	**0.95**	0.79	0.95	0.95	**0.95**
zul-Sadilar	0.07	0.50	0.51	0.51	**0.75**	0.93	0.95	**0.94**	0.74	0.92	0.95	**0.94**
Macroaverage	0.09	0.32	0.64	0.39	**0.81**	**0.91**	**0.91**	**0.91**	0.78	0.89	0.90	0.90

Table 3: Evaluation of the two model architectures - our CNN-LSTM architecture and the TüSeg architecture, trained on each resource, evaluated on data sampled from the corresponding resources. WAcc = word accuracy, MBPrec = morph boundary precision, MBRec = morph boundary recall, MBF = Morph boundary F1). The resource are listed in format [iso 639-3 language code]-[resource name/code].

ing. Since invalid outputs occur rarely, we simply split the word on characters tagged E or S.[1].

[1]The training and evaluation pipeline is available at

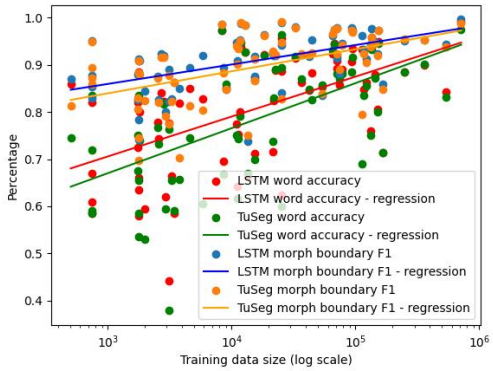


Figure 1: Effect of training data size on model performance. The training size is shown in logarithmic scale

As a simple baseline, we have trained a semi-supervised morphological segmentation model using Morfessor 2.0 (Smit et al., 2014). Some of our languages are very low-resource, and introducing corpora of varying sizes would confound comparability both across the baseline models and between baseline and neural models. Hence, the Morfessor Baseline models were trained exclusively on word types (lists of unique words without corpus frequencies) from the training sets of our resources, used both as the annotated and the unannotated data.

As mentioned in the previous section, we evaluate the models both on the original resources and on custom evaluation data. The results of the evaluation on the training resources are in Table 3. Both architectures achieved high word accuracy and morph-boundary F-measure. On (macro)average, our CNN-LSTM architecture performed slightly better than TüSeg in all the metrics, achieving 81 % word accuracy and 90.6 % morph boundary F1, as compared to 78 % and 89.5 % respectively for TüSeg. Both the architectures have vastly outperformed the Morfessor baseline on all the languages.

The performance gap between the two architectures in terms of word accuracy appears to diminish with growing size of training data, with the CNN-LSTM architecture being better for smaller datasets (see Figure 1). The observed effect of training data size, while consistent with observe linear improvement in performance with exponential increase of training data, as observed by (Olbrich and Žabokrtský, 2025), is overshadowed by the variability caused by the resource- and language-dependent factors.

While the evaluation on training resources yields promising numbers, there is a striking discrepancy between these and the values achieved while evaluating on the manually segmented data (see Table 4), where both the architectures only achieved (macro)average word accuracy of approximately 40 %.

6. Discussion

While it seems that the CNN-LSTM architecture outperforms the TüSeg architecture, some of the largest differences between the two are artifacts. For example, the German *MorphyNet* allows and consistently uses multi-word stems. The CNN-LSTM architecture allows multi-word stems, while TüSeg does not. Morphemes spanning multiple orthographical words are rare, if indeed they exist. In this particular case, the TüSeg is in fact the better segmenter, although nominally it is the worse one. Similarly, for the Korean data, many apparent errors are caused by treating the dashes by default as morpheme boundaries. While this effect is most visible for these two resources, it may on a lesser scale appear in others as well. In some cases, the apparent errors are in fact improvements. For example, *unfailingly* is segmented as *un+failing+ly* in the training data (*eng-MorphoChallenge*), but both the TüSeg and the CNN-LSTM model unfailingly segmented it as *un+fail+ing+ly*.

With respect to the large difference between evaluation measures on the original resources and evaluation on gold data, a large part of the difference is grounded in differences between gold segmentation and segmentation, as presented in the resources (for an overview, see Table 4). Here follows an overview of several types of data incompatibilities and corresponding errors.

- Unrepresentativeness of the training data, leading to false generalizations. For example, *SlavickovaDict* includes only Czech verbs in a slightly archaic infinitive form. As a consequence, in the training data, the last two characters always consist of the Czech inflective affix *-ti*. As a result, the models trained on this dataset tend to segment off the last two characters.

- Incomplete segmentation in the training data. It is noteworthy that the macroaverage precision is significantly higher than the macroaverage recall. For no model is recall more than 10 % higher than precision, while in several cases, the precision is more than 30 % higher than recall.

- Errors in the training data, especially in automatically generated data (e.g. *MorphyNet*) and/or resources not originally designed for morphological segmentation (e.g. *DerIvaTario*, *Uniparser*).

https://github.com/johnvojtech/unisegmenter

				CNN-LSTM				TüSeg			
resource	Both	Same	Same (%)	WAcc	MBPrec	MBRec	MBF	WAcc	MBPrec	MBRec	MBF
ces-Morphynet	39	126	31	0.30	0.71	0.31	**0.44**	**0.34**	0.75	0.30	0.43
ces-SlavickovaDict	0	0	–	**0.30**	0.62	0.65	**0.64**	0.21	0.63	0.53	0.58
deu-CELEX	64	40	63	**0.49**	0.92	0.48	**0.63**	0.46	0.91	0.46	0.61
deu-MorphoChallenge	6	6	100	0.41	0.76	0.61	**0.68**	**0.46**	0.84	0.56	0.67
deu-Morphynet	21	8	38	**0.38**	0.82	0.32	**0.46**	**0.38**	0.83	0.31	0.45
ell-GreekAnnotatedDict	28	12	43	**0.33**	0.55	0.46	**0.50**	0.22	0.53	0.40	0.46
eng-CELEX	121	103	85	**0.71**	0.61	0.36	**0.45**	0.70	0.73	0.32	0.44
eng-MorphoChallenge	4	4	100	**0.77**	0.64	0.63	0.64	0.74	0.64	0.65	**0.65**
eng-MorphoLex	165	143	87	**0.84**	0.74	0.76	0.75	0.82	0.78	0.75	**0.76**
eng-Morphynet	40	26	65	**0.76**	0.75	0.56	**0.64**	0.72	0.72	0.55	0.62
fra-demonette	33	7	21	**0.35**	0.61	0.11	**0.18**	0.34	0.67	0.09	0.16
fra-MorphoLex	69	39	57	**0.45**	0.78	0.38	**0.51**	**0.45**	0.88	0.36	**0.51**
fra-Morphynet	44	11	25	0.36	0.64	0.22	0.33	**0.37**	0.70	0.25	**0.37**
ita-DerIvaTario	3	0	0	0.14	0.37	0.21	**0.27**	**0.15**	0.37	0.17	0.24
ita-Morphynet	25	1	4	0.16	0.47	0.05	**0.10**	**0.17**	0.65	0.05	0.09
lat-WFL	73	44	60	**0.46**	0.83	0.56	**0.67**	0.42	0.86	0.53	0.66
pol-Morphynet	27	1	3	**0.16**	0.70	0.14	**0.23**	**0.16**	0.71	0.13	0.22
rus-Crosslexica	24	11	46	0.38	0.84	0.64	0.72	**0.40**	0.84	0.66	**0.74**
rus-KuznetsEfremDict	59	40	68	**0.49**	0.82	0.75	0.78	**0.49**	0.84	0.75	**0.79**
rus-Morphynet	26	1	4	**0.15**	0.74	0.09	0.15	0.14	0.70	0.10	**0.17**
rus-Tikhonov	51	33	65	**0.45**	0.88	0.59	0.70	**0.45**	0.90	0.59	**0.71**
spa-MATS	60	35	0.58	0.44	0.71	0.58	**0.64**	**0.45**	0.71	0.55	0.62
spa-Morphynet	48	10	0.21	**0.25**	0.34	0.10	**0.16**	0.23	0.33	0.11	**0.16**
Macroaverage				**0.42**	0.69	0.42	**0.49**	0.40	0.72	0.40	0.48

Table 4: Evaluation of the neural segmenters on gold data. Both = how many words are both in training data and in the golden test set. Same=number of words in both train and test set that are segmented in the same way. WAcc = word accuracy, MBPrec = morph boundary precision, MBRec = morph boundary recall, MBF = Morph boundary F1

- Incompatible annotation decisions. For example, in cases of character reduplication (like *Tomm-y* and *Tom-my*), it is unclear whether to increase allomorphy of stems or of suffixes. Our test data as a rule increase allomorphy of stems (i.e., the segmentation would be *Tomm-y*), which doesn't necessarily hold in other resources. Also, some morpheme boundaries are more debateable than others, and the authors of the original resources might have been generally less prone to segment (e.g. the *GreekAnnotatedDictionary*).

Morphological complexity does play a significant role in the performance, as witnessed by the high results for English (even in the case of the fairly inaccurate English MorphyNet). Nevertheless, it seems that typological features impose smaller limitations on the results than size and quality of the training data. For example, results for Russian are consistently high, even though Russian is very morphologically complex.

7. Conclusions

In this paper, we have presented the largest morphological segmentation experiment to date, covering 58 languages. We have trained two state-of-the-art neural models on surface morphological segmentation, achieving 81 % average word accuracy on the original resources. The models can be further used in automatically annotating text corpora on the morpheme level, which is so far very rare. However, the much worse performance on our custom test sets points to the need for further harmonization of morphological segmentation approaches. There is also a large scope for further research in the technical side of the experiment, as both the architecture and training can be potentially improved. For example, if there are multiple inconsistent resources for a language, it might be advantageous to (approximately) train a model succesively on all of these, in increasing order of segmentation quality, or use transfer learning for similar languages. This would avoid false generalizations when possible, while not significantly decrease the quality of the segmentation. If multiple models aren't available, it might still be useful to pretrain on a dataset segmented by an unsupervised method. Our preliminary results indicate improvement in precision, but large decrease in this scenario. Apart from experiments with different architecture (perhaps using another architecture from Batsuren et al. (2022), especially a Transformers-based architecture), we might also try to train a joint model for all the languages, to enable cross-lingual transfer across related languages (Olbrich and Žabokrtský, 2025). We would also like to make use of other morphological features present in the datasets (e.g. lemma or part of speech), as well as devise semi-supervised systems using such information for improved morphological segmentation.

Acknowledgements

This work was supported by the Charles University Research Centre program No. 24/SSH/009, by

the Charles University project GA UK No. 101924, by the Czech Science Foundation project No. 26-21822S, by the SVV project No. 260 821, and by the Ministry of Education, Youth and Sports of the Czech Republic, project No. LM2023062. We also thank anonymous reviewers for their useful remarks.

8. Bibliographical References

Khuyagbaatar Batsuren, Gábor Bella, Aryaman Arora, Viktor Martinovic, Kyle Gorman, Zdeněk Žabokrtský, Amarsanaa Ganbold, Šárka Dohnalová, Magda Ševčíková, Kateřina Pelegrinová, Fausto Giunchiglia, Ryan Cotterell, and Ekaterina Vylomova. 2022. The SIGMORPHON 2022 shared task on morpheme segmentation. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 103–116, Seattle, Washington. Association for Computational Linguistics.

Mathias Creutz and Krista Lagus. 2005. Unsupervised morpheme segmentation and morphology induction from text corpora using Morfessor 1.0. In *Unsupervised morpheme segmentation and morphology induction from text corpora using Morfessor 1.0*. Helsinki University of Technology.

Marie-Catherine de Marneffe, Christopher Manning, Joakim Nivre, and Daniel Zeman. 2021. Universal Dependencies. *Computational Linguistics*, 47(2):255–308.

Leander Girrbach. 2022. SIGMORPHON 2022 shared task on morpheme segmentation submission description: Sequence labelling for word-level morpheme segmentation. In *Proceedings of the 19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 124–130, Seattle, Washington. Association for Computational Linguistics.

Martin Haspelmath. 2020. The morph as a minimal linguistic form. *Morphology*, 30(2):117–134.

Charles F Hockett. 1947. Problems of morphemic analysis. *Language*, 23(4):321–343.

Ursula Klenk and Hagen Langer. 1989. Morphological segmentation without a lexicon. *Literary and Linguistic Computing*, 4(4):247–253.

Manuel Mager, Arturo Oncevay, Elisabeth Mager, Katharina Kann, and Thang Vu. 2022. BPE vs. morphological segmentation: A case study on machine translation of four polysynthetic languages. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 961–971, Dublin, Ireland. Association for Computational Linguistics.

Arya D McCarthy, Christo Kirov, Matteo Grella, Amrit Nidhi, Patrick Xia, Kyle Gorman, Ekaterina Vylomova, Sabrina J Mielke, Garrett Nicolai, Miikka Silfverberg, et al. 2020. Unimorph 3.0: Universal morphology. In *Proceedings of the 12th Language Resources and Evaluation Conference*. European Language Resources Association (ELRA).

Seung-Hoon Na. 2015. Conditional random fields for Korean morpheme segmentation and POS tagging. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 14(3):1–16.

Karthik Narasimhan, Regina Barzilay, and Tommi Jaakkola. 2015. An unsupervised method for uncovering morphological chains. *Transactions of the Association for Computational Linguistics*, 3:157–167.

Michal Olbrich and Zdeněk Žabokrtský. 2025. Morphological segmentation with neural networks: Performance effects of architecture, data size, and cross-lingual transfer in seven languages. In *Text, Speech, and Dialogue*, pages 275–286, Cham. Springer Nature Switzerland.

Marko Pranjić, Marko Robnik-Šikonja, and Senja Pollak. 2024. LLMSegm: Surface-level morphological segmentation using large language model. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 10665–10674, Torino, Italia. ELRA and ICCL.

Teemu Ruokolainen, Oskar Kohonen, Sami Virpioja, and Mikko Kurimo. 2014. Painless semi-supervised morphological segmentation using conditional random fields. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers*, pages 84–89, Gothenburg, Sweden. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725.

Peter Smit, Sami Virpioja, Stig-Arne Grönroos, and Mikko Kurimo. 2014. Morfessor 2.0: Toolkit for

statistical morphological segmentation. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 21–24, Gothenburg, Sweden. Association for Computational Linguistics.

9. Language Resource References

Shaxlo Ashurmamatovna Alimardanova. 2024. Representation of Linguistic Differentiation in the "Word-Formation Dictionary of the Russian Language" by A.N. Tikhonov. *American Journal of Language, Literacy and Learning in STEM Education*, 2(2):463–465.

Ebrahim Ansari, Zdeněk Žabokrtský, Hamid Haghdoost, and Mahshid Nikravesh. 2019. Persian Morphologically Segmented Lexicon 0.5. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Timofey Arkhangelskiy, Oleg Belyaev, and Arseniy Vydrin. 2012. The creation of large-scale annotated corpora of minority languages using UniParser and the EANC platform. In *Proceedings of COLING 2012: Posters*, pages 83–92, Mumbai, India. The COLING 2012 Organizing Committee.

Baayen, R. Harald and Piepenbrock, Richard and Gulikers, Leon. 1995. *The CELEX Lexical Database (CD-ROM)*. University of Pennsylvania. Linguistic Data Consortium, ISLRN 204-698-863-053-1. Catalogue No. LDC96L14.

Khuyagbaatar Batsuren, Gábor Bella, and Fausto Giunchiglia. 2021. MorphyNet: a large multilingual database of derivational and inflectional morphology. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 39–48, Online. Association for Computational Linguistics.

Igor Bolshakov. 2013. Crosslexica: a universe of links between Russian words. *Business Informatics*, 7(3):19–26.

Ramy Eskander, Francesca Callejas, Elizabeth Nichols, Judith Klavans, and Smaranda Muresan. 2020. MorphAGram, evaluation and framework for unsupervised morphological segmentation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 7112–7122, Marseille, France. European Language Resources Association.

Alba Táboas García, Piotr Przybyła, and Leo Wanner. 2025. Exploring morphology-aware tokenization: A case study on Spanish language modeling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 30505–30518, Suzhou, China. Association for Computational Linguistics.

Tanja Gaustad and Cindy A. McKellar. 2024. Updated Morphologically Annotated Corpora for 9 South African Languages. *Journal of Open Humanities Data*, 10(38):1–5.

Abudouwaili Gulinigeer, Abiderexiti Kahaerjiang, Wushouer Jiamila, Shen Yunfei, Maimaitimin Turenisha, and Yibulayin Tuergen. 2021. Morphological analysis corpus construction of Uyghur. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1076–1086, Huhhot, China. Chinese Information Processing Society of China.

Na-Rae Han. 2004. Morphologically Annotated Korean Text LDC2004T03. Web Download. ISBN 1-58563-284-8.

Nabil Hathout and Fiammetta Namer. 2014. Démonette, a French derivational morpho-semantic network. *Linguistic Issues in Language Technology*, 11(5).

Vojtěch John, Zdeněk Žabokrtský, Benjamin Reeves, Magda Ševčíková, Haridanmu Abdukerim, Abduhalik Abduwaiti, Abdukerim Abliz, Ebrahim Ansari, Timofey Arkhangelskiy, Lizaveta Astapenka, Niyati Bafna, Khuyagbaatar Batsuren, Gábor Bella, Pier Marco Bertinetto, Chiara Celata, Hélène Deacon, Konstantinos Diamantopoulos, Šárka Dohnalová, Alexei Fedorenko, Matea Filko, Antonín Forst, Federica Gamba, Timur Garipov, Tanja Gaustad, Fausto Giunchiglia, Anna Glazkova, Hamid Haghdoost, Nabil Hathout, Irina Khomchenkova, Victoria Khurshudyan, Maria Klyucheva, Lukáš Kyjánek, Eleonora Maria Litta, Olga Lyashevskaya, Joël Macoir, Hugo Mailhot, Sun Maosong, Cindy McKellar, Maria Medvedeva, Dmitry Morozov, S.N. Muralikrishna, Fiametta Namer, Anna Nedoluzhko, Mahshid Nikravesh, Aleš Manuel Papáček, Marco Passarotti, Alexey Polyakov, Mihail Potapov, Mishra Pruthwik, Chrysanthi Raftopoulou, Ashwath Rao, Husain Samar, Claudia Sánchez-Gutiérrez, Iurii Savelev-Galiaminskii, Dipti Misra Sharma, Eleonora Slavíčková, Abishek Stephen, Emil Svoboda, Krešimir Šojat, Vanja Štefanec, Luigi Talamo, Jonáš Vidra, Arseniy Vydrin, Maximiliano A. Wilson, Liu Yang, and Aigul Zakirova. 2026. Universal Segmentations 1.5 (UniSegments 1.5). http://hdl.handle.net/11234/1-6130

LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University, Prague.

Mikko Kurimo, Sami Virpioja, Ville Turunen, and Krista Lagus. 2010. Morpho Challenge Competition 2005–2010: Evaluations and Results. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, SIGMORPHON '10, page 87–95, USA. Association for Computational Linguistics.

A.I. Kuznetsova and T.F. Efremova. 1986. *Dictionary of Morphemes of the Russian Language*. Firebird Publications, Incorporated.

Eleonora Litta Modignani, Greta Franzini, Rachele Sprugnoli, Giovanni Moretti, and Marco Passarotti. 2023. CIRCSE/WFL: WFL 1.0.1.

LTRC. 2018. KCIS dependency and coreference annotated corpora. Web Download. Annotation funded by KCIS, DeiTY, Govt. of India. Dependency annotation follows the Paninian Grammar Framework.

Hugo Mailhot, Maximiliano A. Wilson, Joël Macoir, S. Hélène Deacon, and Claudia Sánchez-Gutiérrez. 2020. MorphoLex-FR: A derivational morphological database for 38,840 French words. *Behavior Research Methods*, 52:1008–1025.

Dmitry Morozov, Lizaveta Astapenka, Anna Glazkova, Timur Garipov, and Olga Lyashevskaya. 2025. BERT-like models for Slavic morpheme segmentation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6795–6815, Vienna, Austria. Association for Computational Linguistics.

Claudia H. Sánchez-Gutiérrez, Hugo Mailhot, S. Hélene Deacon, and Maximiliano A. Wilson. 2018. MorphoLex: A derivational morphological database for 70,000 English words. *Behavior Research Methods*, 50(4):1568–1580.

Eleonora Slavíčková. 1975. *Retrográdní morfematický slovník češtiny s připojenými inventárními slovníky českých morfémů kořenových, prefixálních a sufixálních*. Academia, Praha. Souběžné názvy v ruštině, angličtině a francouzštině.

Krešimir Šojat, Matea Srebačić, Tin Pavelić, and Marko Tadić. 2014. CroDeriV: A New Resource for Processing Croatian Morphology. In *Proceedings of the Language Resources and Evaluation (LREC-2014)*, volume 14, pages 3366–3370, Reykjavik. Citeseer.

Luigi Talamo, Chiara Celata, and Pier Marco Bertinetto. 2016. DerIvaTario: An annotated lexicon of Italian derivatives. *Word Structure*, 9(1):72–102.

Jennifer Tracey and Stephanie Strassel. 2020. Basic Language Resources for 31 Languages (Plus English): The LORELEI Representative and Incident Language Packs. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6252–6259, Marseille, France. European Language Resources Association.

UI and NTUA. 2021. Greek annotated dictionary with morphological and linguistic features. Dataset distributed by European Language Grid (ELG), downloaded from https://live.european-language-grid.eu/catalogue/lcr/7360.

Donald Winkelman, Cynthia Wolf, Nathan abd Kong, Alexis Therrien, and Taoran Ye. 2026. Morphseg: An efficient and easy-to-use morpheme segmentation library. `https://github.com/TheWelcomer/MorphSeg`. GitHub repository, accessed 22 Feb 2026.

Tilahun Yeshambel, Josiane Mothe, and Yaregal Assabie. 2021. Morphologically Annotated Amharic Text Corpora. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 2349–2355, New York, NY, USA. Association for Computing Machinery.

Zdeněk Žabokrtský, Niyati Bafna, Jan Bodnár, Lukáš Kyjánek, Emil Svoboda, Magda Ševčíková, and Jonáš Vidra. 2022. Towards universal segmentations: Unisegments 1.0. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1137–1149.

DiNoS: Creating a Data-Driven German Noun Phrase Lexicon from Universal Dependencies

Jacob Lee Suchardt[1,*], **Ronja Laarmann-Quante**[2]
[1]Leipzig University & ScaDS.AI Dresden/Leipzig
[2]Ruhr University Bochum, Faculty of Philology, Department of Linguistics
jacob.suchardt@uni-leipzig.de, ronja.laarmann-quante@rub.de

Abstract

To foster investigations of noun phrase (NP) inflection in German at scale, this paper introduces DiNoS (**Di**stributional **No**un **S**tructure), a data-driven lexicon of NP heads, which includes statistical information on the dependents and the morphosyntactic features of their original in-context appearances. We make available the source code for the extraction of NPs from CoNLL-U treebanks, which includes rule-based heuristics to improve feature annotation coverage and ensures a homogeneous lemmatisation strategy across treebanks. While the resulting JSON-based lexicon is suitable for no-code interaction for non-experts, it is further supported by a toolkit for the automatic calculation of, and access to, various statistical overviews. In this paper, we present the heuristics employed to extract NP datasets from the German Universal Dependencies' Hamburg Dependency and GSD treebanks. In addition, we provide a preview of the emerging DiNoS lexica's properties and discuss some implications of noun and determiner word form ambiguity for NP complexity.

Keywords: corpus linguistics, lexicography, Universal Dependencies, German noun phrases

1. Introduction and Related Work

As a fundamental and highly salient construct, nouns, and noun phrases (NP) by extension, are prevalent across many linguistic research fields such as writing research (Ansarifar et al., 2018), L1 and L2 language use (Lan et al., 2022), and lexical complexity prediction (North et al., 2023). At the same time, NPs also pose challenges for non-proficient individuals during production and understanding; exacerbated in morphologically complex languages (e.g., German).

Lexical complexity prediction (LCP) aims to anticipate and explain comprehension issues for, e.g., children, L2 learners, or individuals with aphasia and distinguishes between *absolute* (or: *objective*) and *relative* complexity (North et al., 2023). Objective complexity includes morphosyntactic (here: primarily morphological structure), semantic, and phonological factors. In contrast, relative complexity depends on situational and personal, individual factors, including, but not limited to, the frequency of a given lemma, word form, or ngram. North et al. (2023) summarise that such statistical factors as well as morphosyntactic, psycholinguistic, and contextual features have been established as good LCP predictors. We propose here that, instead of constituting separate factors, morphosyntactic features should be considered in combination with the lemma, word form, and collocational frequencies of individual occurrences to create morphosyntactic subtypes below the level of word forms, i.e. differentiating superficially identical instances such as the word form *Uhr*.NOMINATIVE.SINGULAR 'clock' from *Uhr*.ACCUSATIVE.SINGULAR. Especially in the case of highly inflectional, fusional languages with non-fixed word order, such as German, statistical familiarity with a lemma does not equate understanding of its morphosyntactic attributes and thus its syntactic role in the larger semantic context of an utterance. Accordingly, inflection of NPs in German — where determiners, adjectives, and nouns must agree according to gender, case, number, and definiteness — remains one of the most common error sources across all L2 proficiency levels (Spinner and Juffs, 2008).

To move towards an account of item frequency, informed by syntactical functions, morphological features, and collocations with other lexical items within NPs, this paper lays the groundwork for their data-driven analysis by 1) extracting NPs from two German Universal Dependencies (UD) (Nivre et al., 2020) treebanks, and 2) formatting the extracted NPs into DiNoS (**Di**stributional **No**un **S**tructure) — a lexicographically inspired, custom JSON data structure — which is organised using the nominal NP heads. DiNoS counts occurrences of i) each NP heads' lemma, ii) lemmas' unique word forms, and iii) their collocations with other words (e.g. determiners, adjectives). Furthermore, each entry is enriched with the occurrences' morphosyntactic features and (when applicable) the word form(s) or part-of-speech (POS) classes of the dependent(s).

We acknowledge the related, and near-simultaneous, work by Krsnik and Dobrovoljc (2025), which presents the STARK toolkit for the extraction of (sub)trees from dependency-parsed CoNLL-U files by specifying syntactic patterns. The

*Part of the work was completed while the author was at FLoV, University of Gothenburg.

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 191–201

		Indefinite				Definite			
		Nom	**Gen**	**Dat**	**Acc**	**Nom**	**Gen**	**Dat**	**Acc**
Sing	**Feminine**	eine Uhr	einer Uhr	einer Uhr	eine Uhr	die Uhr	der Uhr	der Uhr	die Uhr
	Masculine	ein Tag	eines Tages	einem Tag	einen Tag	der Tag	des Tages	dem Tag	den Tag
	Neuter	ein Tier	eines Tieres	einem Tier	ein Tier	das Tier	des Tieres	dem Tier	das Tier
Plur	**Feminine**	ø Uhren	ø Uhren	ø Uhren	ø Uhren	die Uhren	der Uhren	den Uhren	die Uhren
	Masculine	ø Tage	ø Tage	ø Tagen	ø Tage	die Tage	der Tage	den Tagen	die Tage
	Neuter	ø Tiere	ø Tiere	ø Tieren	ø Tiere	die Tiere	der Tiere	den Tieren	die Tiere

Table 1: German article and noun declensions by gender, case, number, and definiteness for *ein/der* + *Uhr/Tag/Tier* ('a/the clock/day/animal').

key difference from our work is that STARK focuses on abstract syntactical patterns and their frequency (e.g. DET <det NOUN), while the highest level of organisation in DiNoS is the NP head's lemma. Additionally, even when the remaining dependents are abstracted, lemma and word form information, as well as morphosyntactical features, are retained in DiNoS. Another UD/CoNLL-U-compatible toolkit, UDeasy (Brigada Villa, 2022), focuses on streamlining data interaction for non-technical users by replacing the need for code-based treebank queries with a graphical interface. The toolkit also lets the user compile an overview of feature distributions, but only on the POS-node level (e.g. "verb" or "noun") or their crosstabs, as opposed to lemma or word form levels.

Following an introduction of NP inflection in German (2) and the selected treebanks (3), we first detail the process of NP dataset creation and feature annotation restoration (4) and proceed with an illustration of the DiNoS conversion procedure and design choices, followed by an overview of the feature distributions (5). Finally, taking the challenge NP inflection presents for German L2 speakers as inspiration, we use the DiNoS toolkit to perform an exemplary analysis of German NPs with definite determiners to highlight form-based morphosyntactic ambiguity from a statistical perspective (6). Our main contributions consist of:[1]

1) DiNoS-lexica of UD_HDT (from Hamburg Dependency Treebank, Borges Völker et al., 2019) and UD_GSD (from GSD, McDonald et al., 2013);
2) Curation of HDT- and GSD-NP datasets, spanning 722k and 49k NPs, respectively;
3) Development of the UD-compatible DiNoS toolkit to extract NP datasets from CoNLL-U treebanks and subsequently create a DiNoS lexicon.

2. Background

Inflection of German NPs depends on the aspects of *gender* ∈ {feminine FEM, masculine MASC, neuter NEUT}, *case* ∈ {nominative NOM, genitive GEN, dative DAT, accusative ACC}, *number* ∈ {singular SING, plural PLUR}, and *definiteness* ∈ {definite DEF, indefinite INDEF} (Table 1). Gender is typically regarded as lexically inherent to a nominal lemma (cf. Opitz and Pechmann, 2016) but does not necessarily conform to its semantic gender. Moreover, inanimate items lacking real-world gender can be morphologically gendered otherwise (Bender et al., 2011). It is often posited that gender cannot be reliably deduced from a word's surface form. Aside from cues of morphological (e.g., derivational suffixes), semantic, or phonological nature, which have been shown to pattern with gender to an extent (Aikhenvald, 2000, referenced in Spinner and Juffs, 2008), only the recent research by Fedden et al. (2025) presented strong evidence that regularities may be more widespread than previously assumed.

Nouns inflect only for case and number. Case marks their syntactical function and is induced syntactically, often by a governing verb or preposition. Number and definiteness are primarily determined by the semantic context, though they can also be analysed as being imposed by the noun's determiner. However, Note the high degree of syncretism in Table 1: A noun's inflection is often not visibly marked on its surface word form and even most article forms are shared across the paradigm and between genders. An NP's surface form may thus often encode different combinations of morphological feature values. Further, the degree of syncretism in a noun's declension depends not only on its gender (e.g., four unique surface forms for *Tag* and *Tier*, vs. only two for *Uhr*) and phonotactical structure (e.g., plural suffix *-En*), but for masculine and neuter nouns, it also depends on their inherent inflectional type (strong, weak, or mixed). Inflectional suffixes are not gender-specific (e.g., *-es* marks GEN.SG.MASC and GEN.SG.NEUT). Moreover, Eisenberg (2020) claims that current language change in German tends to omit inflectional suffixes on nouns (typically in dative, but also seen on accusative and occasionally singular genitive forms), i.e. language use is shifting towards a preference for the weak, more ambiguous inflectional patterns. In conclusion, morphosyntactic cues are

[1] Code on GitHub, data releases of HDT-NP/-DiNoS and GSD-NP/-DiNoS on Zenodo.

ID	Form	Lemma	UPOS	XPOS	Features	Head	Deprel
11-12	im	_	_	_	_	_	_
11	in	in	ADP	APPR	_	13	case
12	dem	der	DET	ART	Case=Dat | Definite=Def | Gender=Masc | Number=Sing | PronType=Art	13	det
13	Wald	Wald	NOUN	NN	Case=Dat | Gender=Masc | Number=Sing	14	obl
11	im	_	_	APPRART	Case=Dat | Definite=Def | Gender=Masc | Number=Sing | PronType=Art	13	_
13	Wald	Wald	NOUN	NN	Case=Dat | Gender=Masc | Number=Sing	14	obl

Table 2: Example CoNLL-U sentence dev-s26 (GSD) (en. 'in the forest') before (above) and after (below) APPRART restoration.

not consistently found on nominal word forms and also tend to be ambiguous. As a consequence, a noun's encoded morphosyntactic features and thus semantic role often cannot be extrapolated from a noun's form alone.

At times, the ambiguity is reduced by the noun's dependents, such as the highly informative determiners (e.g., *dem* vs. *den* disambiguate Tag.DAT.SG.DEF and Tag.ACC.SG.DEF), but many forms are still ambiguous (e.g., *die Uhr* can be nominative or accusative). In total, there are only six distinct definite (die, der, des, dem, den, das) and indefinite (ein, eine, einer, eines, einem, einen) article word forms, most of which are also shared across genders. Notably, articles do not occur with indefinite plural nouns (ø *Uhren*.DEM.PL.IND and stop inflecting for gender in the definite plural condition.

3. Resources

To establish a proof of concept and an initial database, we select two pre-annotated treebanks from the Universal Dependencies (UD) (Nivre et al., 2020; de Marneffe et al., 2021) 2.15 collection (Zeman et al., 2024). UD is a communal effort to create treebanks with cross-linguistically applicable annotations and guidelines. It encompasses syntactic dependencies, part-of-speech tags, and morphosyntactic features. Due to the challenging nature of morphosyntactic parsing of German (Do, 2023), we focus on large, pre-annotated and, at least in part, quality-controlled data sources to minimise noise in our results.

The bulk of our data stems from UD_German-HDT (UD-HDT), one of the largest German treebanks, which is a subset of the Hamburg Dependency Treebank (Foth et al., 2014) that was converted to UD using TrUDucer (Hennig and Köhn, 2017; Borges Völker et al., 2019). Along with the conversion, annotation was also expanded with external sources and manual input. The data had been annotated manually and checked for consistency using DECCA, except for one half of the training data (Borges Völker et al., 2019). Thematically, HDT consists of editorial, political, and legal articles related to computer science and technology, sourced from the German news site heise.de (1996–2001).

The second treebank is UD_German-GSD (UD-GSD), first released by McDonald et al. (2013) who stated that the domains consist of news texts from the Tiger Treebank (Brants et al., 2004) and unspecified consumer reviews. Starting with the UD v2.0 release, duplicates were removed from the data and texts from the "web domain" were added to the training data, supposedly from Wikipedia among other sources. HDT surpasses GSD in size, sentence construction complexity, and vocabulary specificity. GSD is characterised by a more colloquial register, opening up the possibility of investigating domain effects.

In both treebanks, we pool the data from all train, dev, and test files. Sentences are pre-split, tokenised, and given in the CoNLL-U format (e.g., Table 2) which we process using the conllu library. Words are further annotated with a sentence position ID, word form, lemma, universal POS tag (UPOS), a language-specific POS tag (XPOS, here: Stuttgart-Tübingen-TagSet (STTS, Schiller et al., 1999), morphological features, the governing head token's ID, and the dependency relation (deprel) to that head token. Together, the data amount to 206k sentences (HDT: 189.9k, GSD: 15.6k) consisting of a total of 3.8M tokens (HDT: 3.5M, GSD: 268.4k).

4. Noun Phrase Dataset Creation

To extract the NPs from a CoNLL-U sentence, we first select all its NP heads and then add each head's direct, agreeing dependents. We restrict our choice of NP heads to common nouns with a minimum length of two letters,[2] and to simple NPs (nouns with determiners, prepositions,[3] or adjectival/adverbial modifiers). For composite and multiword nouns (e.g., *ein Karate Meister* 'a karate master'), the subordinate modifier (here: *Karate*) does not constitute an independent NP and is instead extracted along with the article as a dependent of *Meister*. The other dependents were retrieved via their positional ID and head token ID annotations.

[2]No regular common noun word form of German consists of less than two letters, although such tokens may exist in the data.

[3]Following UD annotation, we consider Prepositional Phrases as a subtype of NP.

Thus, complex NPs such as those involving subordinate clauses or instances like *von der Auswahl und der Qualität* 'by the selection and quality', where *Qualität* has a *conj* relation to *Auswahl*, are broken up into two independent NPs: *von der Auswahl* and *der Qualität*.

In this way, we extract about 722k eligible NPs from HDT, and 49k NPs from GSD. The number of NPs excluded due to insufficient NP head length was low (HDT: 614, GSD: 218).

4.1. Feature Restoration

Initial inspection revealed that about 1.32% of NP heads in GSD and 89.46% of NP heads in HDT had insufficient annotation for gender, case, or number.[4] We attempt to restore this annotation on NP heads and determiners during NP extraction by searching the feature annotations of a given NP's grammatically agreeing dependents for the missing values and copying them. Additionally, we infer missing case values from the NP head's syntactic function, provided it has a definite mapping to a case value, i.e. nsubj→nom, obj→acc.

Feature restoration was successful for nearly 75% of incomplete HDT items. This leaves about 23.7% NPs in HDT-NP that remain underspecified in one or more features. In contrast, gaps in the annotation were comparatively rare in GSD (<1.5%).

4.2. Token Contraction

According to UD guidelines, tokens corresponding to the XPOS tag APPRART (STTS: "preposition with fused article") were split into two separate tokens in both treebanks: APPR (UPOS: ADP) and ART (UPOS: DET), e.g., *im* → *in dem*. However, this split into *syntactic* words introduces additional ART/DET tokens and does not accurately depict the *orthographic* words used in the source materials. Therefore, we restore the original contracted forms from the preserved APPRART relic tokens (Table 2). APPRART relics are identifiable by their position ID's unique format (a range of IDs encompassing the respective deconstructed forms' position IDs). We use the XPOS tag APPRART and copy the deconstructed adposition's position ID and head token ID, as well as the unified feature annotation of both the adposition and determiner. Lastly, the now redundant deconstructed tokens are dropped from the NP. In total, 53,526 APPRART tokens were reconstructed across NPs from HDT and 3,649 across GSD.

Ultimately, we obtain two NP datasets: **HDT-NP** (722,135 NPs, 1,684,425 tokens, avg. 2.33 tokens/NP) and **GSD-NP** (49,425 NPs, 118,988 tokens, avg. 2.41 tokens/NP — NP subsets of the CoNLL-U UD treebanks with improved annotational coverage.

[4] In the following, we focus on these three features and exclude definiteness since nouns do not inflect for this feature and its expression on NPs is unequivocal.

5. Distributional Noun Structure

Our next goal is to transform the extracted and restored NPs from each treebank into the custom DiNoS lexicon format. For each lemma, we aim to capture absolute frequencies of the lemma and its word forms, as well as information on the morphosyntactic features and collocations associated with each form's occurrence (Figure 1).

5.1. Relemmatisation Algorithm

GSD and HDT exhibit different lemmatisation strategies for nominal compounds — a highly productive construction in German, found among many colloquial and lexical items, e.g. *Krankenhaus* 'hospital' (sick.NOUN + house.NOUN). While compounds receive independent lemmas in GSD, their counterparts in HDT are tagged with the head of their morphological lemmas (i.e. *Haus* 'house'). Due to the productivity and frequent lexicalisations of these constructions, as well as the difference in morphological complexity of compounds and atomic nouns, we opt to apply GSD's lemmatisation strategy to HDT using a custom form- and rule-based relemmatisation algorithm.

Using the lemma of a compound's morphological head, we infer the compound's uninflected form and instantiate a new lemma entry: In the simplest case, the compound form is not inflected, e.g. *Krankenhaus* is a direct match for the lemma *Haus*. Alternatively, an inflected compound form *Krankenhäuser*.NEUT.NOM.PL can be lemmatised by either leveraging the large size of the treebanks and finding a match of an inflected non-compound from (given: *Häuser* → *Haus*; therefore: *Krankenhäuser* → *Krankenhaus*); or, if no matching inflection was found, by establishing a match with the morphological head lemma after reverting the plural umlaut *ä* to *a*: *Krankenhauser* → *Krankenhauser* → *Krankenhaus*. After all newly established compound lemmas were added, forms which still could not be matched to a lemma receive the pre-existing 'Unknown' lemma. At this time, we prune NPs whose head lemma is 'Unknown' or purely numerical.

Table 3 reports the number of total NPs and the therein contained unique lemmas and unique word forms, as well as the the ratio of unique word forms to lemmas at three stages: Extracted NPs prior to relemmatisation (*raw*), post-relemmatisation but excluding the pruning of 'Unknown' lemma and numerical items (*noisy*), and post-relemmatisation

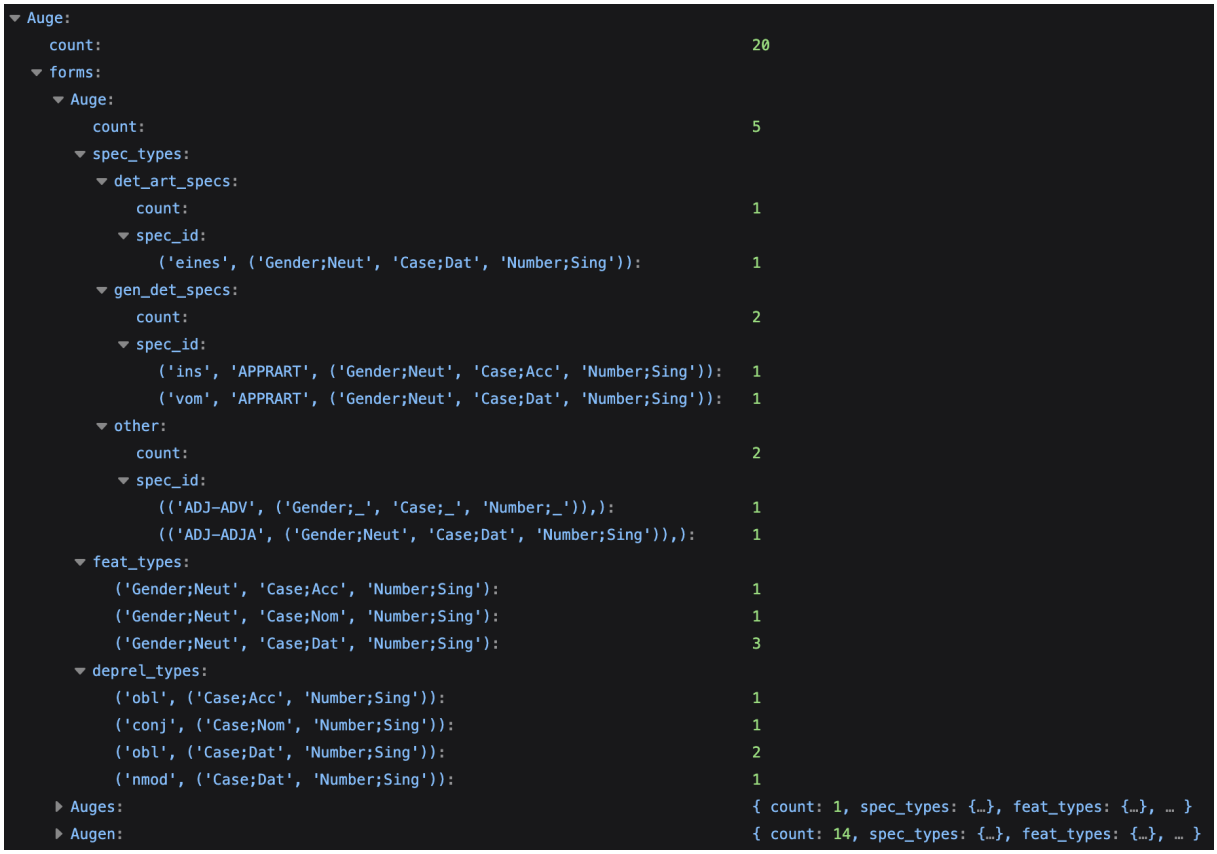


Figure 1: Demonstration of DiNoS format with word form entry *Auge* (lemma *Auge* 'eye') from GSD-DiNoS.

with pruning (*clean*/DiNoS).

		NPs	Lemmas	Forms	Ratio
HDT-NP	raw	722,135	13,777	116,019	8.42
	noisy	722,135	84,689	110,986	1.31
	clean	707,706	84,598	102,418	1.21
GSD-NP	raw	49,425	17,464	20,305	1.163
	noisy	49,425	17,435	20,195	1.158
	clean	49,416	17,433	20,190	1.158

Table 3: Effects of relemmatisation and pruning.

The coarse lemmatisation of HDT is clearly visible in the raw data. Despite being more than 10 times the size of GSD and having substantially more unique word forms (GSD: 20.3k, HDT: 116k), raw HDT has notably fewer unique lemmas (GSD: 17.5k, HDT: 13.7k). Furthermore, the form to lemma ratio of 8.42 in Table 3 implies that each lemma in HDT exhibits around 8 inflectional forms on average, however, no noun class in German has more than 6 inflectional forms. The DiNoS relemmatisation algorithm aligns the datasets: In the *noisy* row, HDT's number of unique lemmas (84.7k) corresponds much more closely to its overall size, as does the aforementioned ratio (1.21) to GSD's (1.16). Although some NPs with numerical lemmas were dropped from the *clean* row (HDT: 145 NPs, 92 lemmas; GSD: 1 NP, 1 lemma), pruning during relemmatisation was mostly confined to 'Unknown' lemma NP heads (HDT: 20,674 NPs, 21 lemmas; GSD: 8 NPs, 5 lemmas). Additionally, the raw HDT data contained 23,260 NP heads which had already been tagged with the 'Unknown' lemma and of which 72.7% were successfully remapped to a lemma entry by the algorithm.

771 (4.4%) of the lemmas in GSD-DiNoS counted more than 10 occurrences in the data, as is the case for 6549 (7.7%) lemmas in HDT-DiNoS (Figure 2). Owing to its large source treebank, HDT-DiNoS also contains 1011 lemmas which occurred >100 times (mean: 400, median: 211). These items present a start towards collecting data for exploring lemma-level statistics, such as case preferences on the word level.

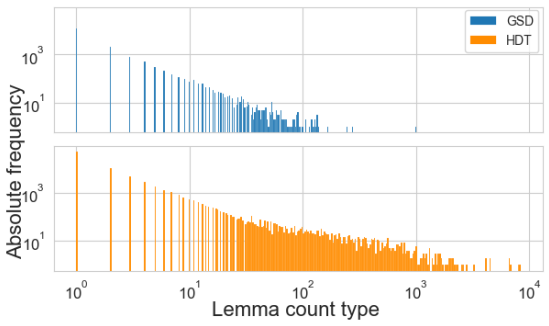


Figure 2: Frequency distribution of lemma counts.

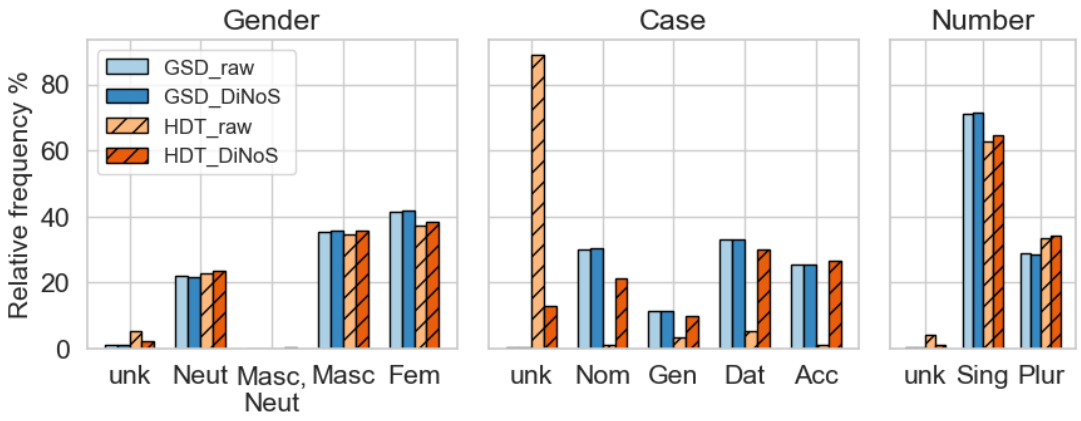


Figure 3: Relative frequency of feature values on NP heads pre/post feature restoration and pruning.

5.2. Structural Design

The subset of extracted and restored NPs from each treebank is transformed into a JSON-based DiNoS lexicon, which aggregates NPs with the same head lemma (Figure 1). For each lemma, absolute frequencies of the lemma (here: 20) and its word forms (here *Auge*: 5, *Auges*: 1, *Augen*: 14) are captured. Moreover, each occurrence feeds into three areas of interest: morphosyntactic features (FEAT(URE)_TYPES) in isolation (gender, case, number), in combination with dependents (SPEC(IFIER)_TYPES), and in combination with the syntactic function (DEP(ENDENCY)REL(ATIONS)_TYPES). From the example in Figure 1, it is visible that the word form *Auge* occurred most often as *Auge*.SG.DAT. The subcategory SPEC_TYPES warrants further explanation: To gain insight into how likely a reader is to encounter a given NP pattern, collocations are tracked as the dependents that co-occur with a given NP head, along with their feature annotation. These specifiers are assigned to one of three subcategories: 1) determiner-articles (DET_ART_SPECS), 2) non-article "general" determiner (e.g. possessive pronouns; including APPRART; GEN_DET_SPECS), 3) OTHER (e.g., adjectives), including no further dependents (e.g., indefinite plural). Since these categories are technically not mutually exclusive, they are assigned hierarchically in the order they were presented above, i.e. presence of an article necessarily ensues category 1), regardless of the presence of other, general determiners. Depending on the subcategories outlined above, occurrences are tracked with the specifier's 1) word form, 2) word form and XPOS tag, or 3) a UPOS-XPOS tag pair (if applicable, else we use a NON_SPEC placeholder). The form *Auge* appeared most commonly in dative case, and tends to occupy indirect object (*obl*) positions.

5.3. Overview of Macro-Feature Distributions

Using the DiNoS toolkit, this section presents the distributions of the morphological features, dependency relations, and frequency of specifier groups across HDT- and GSD-DiNoS. The morphological feature overview also includes the distribution over the raw data, to highlight the effect of the feature restoration. Due to the immense number of lemma and word forms together with combinations of gender, case, number, dependency relations, and specifier groups, we leave many of the possible analyses open for future work.

Morphology In Figure 3 we examine the distributions of morphological features across (1) all initially eligible NPs before any pruning or annotation modification (*raw*), and (2) the clean DiNoS lexica after feature restoration, relemmatisation, and pruning (*DiNoS*).[5] For GSD, the overall coverage of feature annotation in the raw data is high: only for gender are >1% of NP heads missing a value and differences from the DiNoS version are minimal. On the other hand, of the 722k NPs initially extracted from UD-HDT, 5% of NP heads lack gender, 4% lack number, and a striking 89% lack annotation for case. Beginning with gender and number, our feature restoration heuristics reduce unknown values by -1.94 and -2.21, respectively, while pruning further lowers them down to 2.27% (gender) and 0.91% (number) in the final version. For case annotation, look-up from dependents and inference from dependency relations was highly successful, reducing unknown case items by -76.03 down to 13.08%, and 12.85% after pruning. The restoration of dative case in particular indicates that feature annotation in HDT is sparsely distributed among

[5]Since pruning had minimal effects on the distributions in either dataset, we omit the intermediary stage of pruning without restoration/vice versa from the graphic.

agreeing constituents, rather than incomplete: NP heads are annotated for gender and number, while case is situated on dependents such as determiners.

Generally, distributions between the DiNoS lexica are similar for gender and number: The most common gender is feminine (avg. 40.1%), followed by masculine (avg. 35.6%) and neuter (22.6%) — error margins of ∼2 points withstanding. These values broadly correspond to distributions of gender over the most common nominal lemmas in German reported in Fedden et al. (2025), although highly frequent lemmas have a higher weight in the data-driven feature distribution here. The extent of alignment between the lexica is less clear for grammatical case since 12.85% of items from HDT are still underspecified for case. At 9.2%, the largest gap currently persists for nominative case (HDT: 21.07%, GSD: 30.22%). Aside from nominative, which is the second most common case in GSD and third most common in HDT, the frequency ordering and values of the remaining cases align so far: dative (avg. 31.45%, ±1.58), accusative (25.91%, ±0.64), genitive (10.46%, ±0.8). Regarding grammatical number, singular NP heads account for about two thirds of the data. Plural is favoured more strongly in HDT than GSD, which may be in part attributed to the fact that the most common NP heads consist of mainly plural nouns (e.g., counting/currency units such as *Prozent* 'percent') and form a Zipfian distribution.

Dependency Relations Table 4 shows the 8 most common dependency relation tags found on the NPs underlying HDT- and GSD-DiNoS, each covering >95% of all instances per lexicon.

	nmod	obl	nsubj	obj	conj	nsubj:pass	root	appos
GSD	25.5	23.8	15.1	11.6	9.1	4.0	3.7	2.1
HDT	23.2	25.4	16.5	17.2	6.2	3.1	2.4	3.5

Table 4: Top 8 most common deprel tags.

Specifiers Nearly half the NPs in HDT- and GSD-DiNoS contain an article (*ein/der*), around 16% contain another type of (non-article) determiner, 22% occur with non-determiners (e.g. adjectives), and about 13% of NPs consist of just the NP head without any direct dependents (Figure 4)

6. Example: Morphological Subtypes of Definite Articles

In the following, we focus on the most common type of NP: NPs with a definite article (HDT-NP: 38.17% of all NPs, GSD-NP 39.38%). First, we explore how the morphological features of this subset differ from the overall population of NPs. Second, we take a closer look at the distribution of morphological features over the definite article's word forms, examine which morphological subtypes reveal themselves, and how these may compete with each other in terms of frequency and ambiguity.

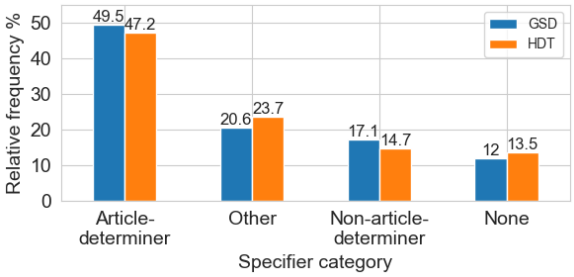


Figure 4: Relative frequency of specifier categories.

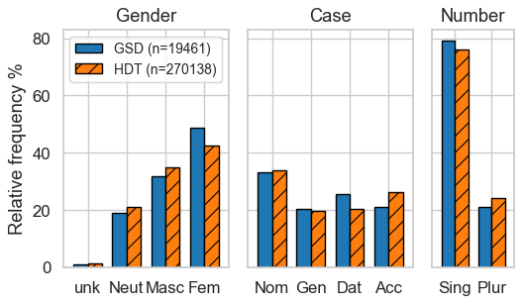


Figure 5: Relative frequency of feature values on NPs with definite articles.

The morphological feature distributions in Figure 5 roughly follow the patterns seen in their supersets (Figure 3), while only very few annotations (exclusively gender) remain underspecified. However, feminine gender is notably more common (>+5%) in these subsets; the increase in each exceeding the possible margin of error from previously unknown gender values. Within grammatical number, the shift and proclivity towards singular is also substantial (avg. +9.47). Furthermore, case values align: Nominative is the most common case in both (∼33.5%) and genitive (∼19.9%) is also more frequent in the definite article subsets. Still, there appear to be population differences: With a gap of ±4.87, dative is favoured by GSD, while HDT favours accusative (±5.12). Therefore, although similarities persist, there are also some discrepancies between the feature distributions across definite article NPs and those seen across all NPs.

As seen in Section 2, definite articles may aid in deciphering an NP's morphological features due to the more pronounced inflectional markings, thus, in turn, aiding language processing (Rogers and Gries, 2022). Figure 6 reveals how the morphological feature values from Figure 5 are distributed

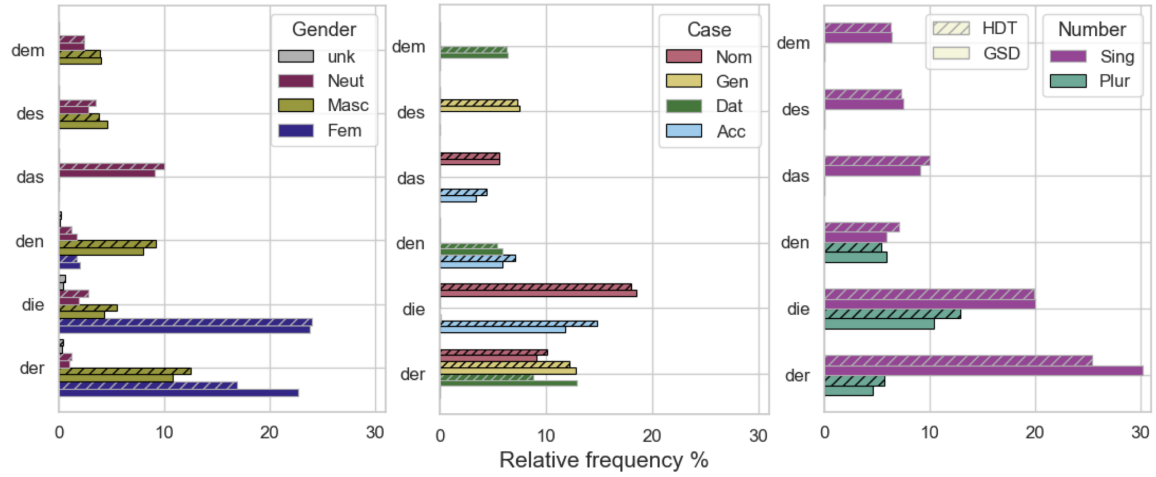


Figure 6: Relative frequency of morphological features across the definite article's distinct word forms.

across the definite article's word forms. Patterns of cumulative relative frequency and degrees of ambiguity emerge: The less common forms *das*, *des*, and *dem* (<10% of all definite article occurrences) display ambiguity in only one cardinal feature each (*das*: NOM/ACC case, *des*/*dem*: MASC/NEUT gender), but their competing values occur at similar rates. *Den* is a slightly more common form (∼10%) but also subject to more syncretism. The two most common forms by far, *der* (avg. 33%) and *die* (avg. 32%), ultimately account for about two thirds of all definite article instances together. *Die* has the same degrees of ambiguity as *den* (all genders and numbers), but appears only in either nominative or accusative case. *Der* appears in one more grammatical case — all but accusative. Both of these highly frequent forms encode feminine and singular most often, which matches the cumulative frequencies of the individual feature values seen in Figure 5.

Next, we go beyond the association of singular features to word forms and view the frequencies of the **morphologically unique subtypes** of the definite articles to explore whether forms have "default interpretations" or whether their subtypes occur at similar rates. Using a tuple of the fully specified gender, case, and number values to distinguish the word forms, 24 unique morphological subtypes emerge in Figure 7 (not including the 4 types with indeterminate gender). In GSD, the most common subtype is *der*.FEM.DAT.SG (12.9% of all definite article instances) so that the most common appearance of *der* in GSD encodes neither NOM case nor MASC gender, as the commonly cited "default" of nominative singular (with a definite article) would demand (Schriefers and Teruel, 2000; Fedden et al., 2025). Nevertheless, *der*.MASC.NOM.SG is also highly frequent at 9.1%. In HDT this type does constitute both the most frequent *der* subtype and the second most frequent definite article subtype in general (10.1%). Overall, the top 5 most frequent and all but one of the bottom 5 least frequent types are all subtypes of *die* or *der*. The subtypes can be split along the number feature, with more frequent singular (≥5%) and low frequent plural (≤5%) subtypes. Another article form whose subtypes are split along the number feature is *den*. The form exhibits an archetypical subtype, *den*.MASC.ACC.SG, which accounts for around half of its instances, and

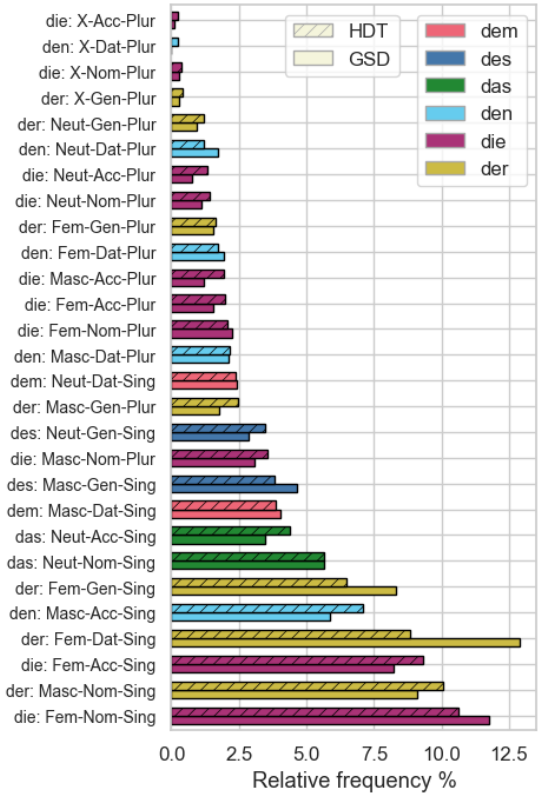


Figure 7: Relative frequency of morphological subtypes of definite articles.

three plural subtypes *den*.DAT.PL with lower, similar frequencies.

In plural, the definite article no longer distinguishes between genders and only changes depending on the grammatical case. Among the plural forms, note that masculine is the most common gender in: dative (*den* subtypes), nominative (*die* subtypes), and genitive case (*der* subtypes), but not in accusative case (*der* subtypes). This contrasts with the observation that feminine is the most common gender of singular subtypes. Conversely, masculine and accusative had been very frequent in HDT-DiNoS overall, but they do not constitute the most common gender-case combination among the plural subtypes of the definite article. It ensues that interaction effects between features are possible.

7. Conclusion and Future Work

We presented DiNoS (Distributional Noun Structure), a lexicon-like data structure and toolkit for the fully automated extraction of NP datasets from pre-annotated German data in the CoNLL-U format. From the UD-HDT and -GSD treebanks, we created the datasets HDT-NP (722.1k NPs, 1.7M tokens) and GSD-NP (49.4k NPs, 119.0k tokens), comprising NPs with common nouns as heads and their direct and grammatically agreeing dependents. As part of this extraction, feature annotation on the NP heads and their determiners was restored from agreeing dependents to maximise coverage of the annotation for gender, case, and number without the need for external knowledge resources, greatly expanding the annotation coverage for HDT in particular. Using DiNoS, we further aligned the lemmatisation granularities of the two NP datasets and transformed them into a data-driven, lexicon-like JSON data structure (HDT-DiNoS: 84.6k lemmas, GSD-DiNoS: 17.4k lemmas) — suitable for both code-based interaction and manual lookup.

DiNoS and its supplied loader allow for the calculation of and access to various frequency distributions of lemmas, word forms, features, dependency relations, and collocational constructions on the corpus, lemma, and word form levels, many of which are pre-computed automatically. As an exemplary analysis, Section 6 showed that the morphological feature distributions of certain subsets of NP constructions, such as NPs with definite articles, differ from those on the population level. Furthermore, we demonstrated that the morphological subtypes of the superficially identical definite article word forms often compete with each other in terms of frequency, building a case for including the degree of competition and ambiguity in assessing, e.g., an NP's complexity.

The analysis example here only scratches the surface of research possible with DiNoS. For example, feature correlations and interactions, such as gender x case, on the noun type or article basis can be further explored (see also Rogers and Gries, 2022). Comparisons of our distributions to related work would be desirable, however, the authors are not aware of any further comparable works. We expect the toolkit to be compatible with other CoNLL-U treebanks of German and potentially adaptable to other languages. In future work, we also aim to expand the data structure design and, although automated morphosyntactc annotation remains challenging in German, to include external tools to verify and expand the morphological annotations.

Limitations

During NP extraction, heads were restricted to common nouns which had to be attested by both its UPOS and XPOS tag. The reason being that, for one, both treebanks contain truncations (e.g. *Tages- und Nachtzeit* 'day- and nighttime', GSD dev-s471), which are only detectable via the XPOS=TRUNC tag (HDT: 90, GSD: 82). They are unsuitable as NP heads due to lacking feature annotation and having no eligible direct dependents.

Our work focused on pre-annotated, quality controlled treebanks to minimize noise. GSD contained 1,375 nouns with visible tagging errors, i.e. inconsistencies where nominal and other POS tags were mixed, which we excluded to minimise noise in our NP datasets. GSD also had slightly higher error rates than HDT in the definite article annotations with 0.22% vs. 0.03% of all definite articles being affected; errors here mean combinations of article forms and feature values that are not admissible in German (e.g., **das*.FEM.NOM.SG). Potential error rates among the extracted items with admissible combinations are unknown and cannot be ruled out entirely. However, there should be sufficiently few, or indeed ungrammatical constructions from unmoderated, web-scraped materials. Errors in HDT would likely stem from the second half of the training data which had reportedly not been checked for annotation errors (Section 3).

In HDT-NP, ∼12.85% of NPs are still missing case annotations, while the only remaining under underspecified article subtypes for HDT in Figure 6 and Figure 7 lack merely gender. By utilising the lemma-based organisation of the DiNoS the remaining unknown values could be reduced further by either referencing the lemma's gender entry. However, the implementation is not as straightforward, since homographs with differing genders may share the same surface forms (e.g. *Band*.FEM '[musical] band' vs. *Band*.MASC 'volume [of a book series]'). This aspect could be improved upon by, for instance, sorting forms under a lemma entry by their associated gender.

8. Bibliographical References

Alexandra Y. Aikhenvald. 2000. *Classifiers: A Typology of Noun Categorization Devices*. Oxford University Press.

Ahmad Ansarifar, Hesamoddin Shahriari, and Reza Pishghadam. 2018. Phrasal complexity in academic writing: A comparison of abstracts written by graduate students and expert writers in applied linguistics. *Journal of English for Academic Purposes*, 31:58–71.

Andrea Bender, Sieghard Beller, and Karl Christoph Klauer. 2011. Grammatical gender in german: A case for linguistic relativity? *Quarterly Journal of Experimental Psychology*, 64(9):1821–1835. PMID: 21740112.

Sabine Brants, Stefanie Dipper, Peter Eisenberg, Silvia Hansen-Schirra, Esther König, Wolfgang Lezius, Christian Rohrer, George Smith, and Hans Uszkoreit. 2004. Tiger: Linguistic interpretation of a german corpus. *Journal of Language and Computation*, 2:597–620.

Bich-Ngoc Do. 2023. *Neural Techniques for German Dependency Parsing*. Ph.D. thesis, Ruprecht-Karls-Universität Heidelberg.

Peter Eisenberg. 2020. *Grundriss der deutschen Grammatik: Das Wort*. J.B. Metzler, Stuttgart.

Sebastian Fedden, Matías Guzmán Naranjo, and Greville Corbett. 2025. Typology meets statistical modeling: The german gender system. *Language*, 101:251–290.

Kilian A. Foth, Arne Köhn, Niels Beuck, and Wolfgang Menzel. 2014. Because size does matter: The Hamburg dependency treebank. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2326–2333, Reykjavik, Iceland. European Language Resources Association (ELRA).

Felix Hennig and Arne Köhn. 2017. Dependency tree transformation with tree transducers. In *Proceedings of the NoDaLiDa 2017 Workshop on Universal Dependencies (UDW 2017)*, pages 58–66, Gothenburg, Sweden. Association for Computational Linguistics.

Ge Lan, Qiusi Zhang, Kyle Lucas, Yachao Sun, and Jie Gao. 2022. A corpus-based investigation on noun phrase complexity in l1 and l2 english writing. *English for Specific Purposes*, 67:4–17.

Kai North, Marcos Zampieri, and Matthew Shardlow. 2023. Lexical complexity prediction: An overview. *ACM Comput. Surv.*, 55(9).

Andreas Opitz and Thomas Pechmann. 2016. Gender features in German: Evidence for underspecification. In *Proceedings of the 7th Tutorial and Research Workshop on Experimental Linguistics*, pages 127–130.

Phillip G. Rogers and Stefan Th. Gries. 2022. Grammatical gender disambiguates syntactically similar nouns. *Entropy*, 24(4).

A. Schiller, S. Teufel, C. Thielen, and C. Stöckert. 1999. Guidelines für das tagging deutscher textcorpora mit stts (kleines und großes tagset).

Herbert Schriefers and Encarna Teruel. 2000. Grammatical gender in noun phrase production: The gender interference effect in german. *Journal of experimental psychology. Learning, memory, and cognition*, 26:1368–77.

Patti Spinner and Alan Juffs. 2008. L2 grammatical gender in a complex morphological system: The case of german. *Iral-international Review of Applied Linguistics in Language Teaching - IRAL-INT REV APPL LINGUIST*, 46:315–348.

9. Language Resource References

Borges Völker, Emanuel and Wendt, Maximilian and Hennig, Felix and Köhn, Arne. 2019. *HDT-UD: A very large Universal Dependencies Treebank for German*. Association for Computational Linguistics.

Luca Brigada Villa. 2022. Udeasy: a tool for querying treebanks in conll-u format. In *Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-10)*, pages 16–19, Marseille, France. European Language Resources Association.

de Marneffe, Marie-Catherine and Manning, Christopher D. and Nivre, Joakim and Zeman, Daniel. 2021. *Universal Dependencies*. MIT Press.

Luka Krsnik and Kaja Dobrovoljc. 2025. STARK: A toolkit for dependency (sub)tree extraction and analysis. In *Proceedings of the 23rd International Workshop on Treebanks and Linguistic Theories (TLT, SyntaxFest 2025)*, pages 44–51, Ljubljana, Slovenia. Association for Computational Linguistics.

McDonald, Ryan and Nivre, Joakim and Quirmbach-Brundage, Yvonne and Goldberg, Yoav and Das, Dipanjan and Ganchev, Kuzman and Hall, Keith and Petrov, Slav and

Zhang, Hao and Täckström, Oscar and Bedini, Claudia and Bertomeu Castelló, Núria and Lee, Jungmee. 2013. *Universal Dependency Annotation for Multilingual Parsing*. Association for Computational Linguistics.

Nivre, Joakim and de Marneffe, Marie-Catherine and Ginter, Filip and Hajič, Jan and Manning, Christopher D. and Pyysalo, Sampo and Schuster, Sebastian and Tyers, Francis and Zeman, Daniel. 2020. *Universal Dependencies v2: An Evergrowing Multilingual Treebank Collection*. European Language Resources Association.

Zeman, Daniel and Nivre, Joakim and Abrams, Mitchell and Ackermann, Elia and Aepli, Noëmi and Aghaei, Hamid and Agić, Željko and Ahmadi, Amir and Ahrenberg, Lars and Ajede, Chika Kennedy and Akhundjanova, Arofat and Akkurt, Furkan and Aleksandravičiūtė, Gabrielė and Alfina, Ika and Algom, Avner and Alnajjar, Khalid and Alzetta, Chiara and Andersen, Erik and Andrews, Matthew., ... Znotiņš, Artūrs. 2024. *Universal Dependencies 2.15*. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL).

Figurative, Polysemous, Conventional: Designing a Dataset of Regular Metaphor

Anna Temerko, Pablo Gamallo, Marcos Garcia
CiTIUS – Research Centre in Intelligent Technologies, Universidade de Santiago de Compostela
Santiago de Compostela, Spain
a.temerko@usc.es, {pablo.gamallo, marcos.garcia.gonzalez}@usc.gal

Abstract

Metaphor, a figure of speech and a cognitive device, offers a powerful way to explain one conceptual domain in terms of another. Particularly successful metaphorical mappings conventionalize through frequent use and lose their creative quality. They become sense extensions of polysemous words. Our dataset project captures such metaphors with ten regular polysemy patterns that manifest repetitively in the meaning structures of English words. Regular metaphor, unlike its counterpart regular metonymy, has not previously received a dedicated dataset, and we intend to close this gap. The dataset under construction features naturalistic sentences extracted from a general language corpus and is manually annotated with sense labels for metaphorically extended polysemes. Its intended use is to support linguistic, cognitive and computational investigations into patterns of meaning in polysemy, while accounting for its complexity, regularity, continuity and heterogeneity. We see neural language models as an excellent experimental ground for such research because they are able to show both distributional (continuous) and symbolic (discrete) behavior in language processing and representation. In this paper, we reflect on how these systems tally.

Keywords: metaphor, polysemy, large language models

1. Introduction

Metaphor is a figure of speech that involves mappings between two conceptual domains based on analogy or similarity between the source domain and the target domain (Lakoff and Johnson, 1980). Through frequent use and entrenchment (Grady et al., 1999; Langacker, 2017), metaphors conventionalize, thus moving away from the creative pole of the continuum (Cardillo et al., 2012). Some metaphors also lose their perceived figurative quality in this process and are processed similarly to literal expressions (Cardillo et al., 2010, 2017; Lai et al., 2009). These shifts are what contributes to a conventionalized, entrenched metaphor becoming a sense extension of a polysemous word (Egg and Kordoni, 2022; Reijnierse et al., 2018; Steen et al., 2010b). Some metaphoric mappings are so salient that they repeat themselves over and over across lexical units of a semantic domain, thus forming regular polysemy patterns (Lombard et al., 2023, 2024; Srinivasan and Rabagliati, 2015).

Metaphor and polysemy present a double challenge for natural language processing: polysemy, a language phenomenon that implies multiplicity of word's senses, causes ambiguity (Haber and Poesio, 2024), whereas metaphor, describing one concept in terms of another, requires semantic cross-domain integration (Momen et al., 2026). Both phenomena increase processing difficulties for humans and machines and both are ubiquitous: most lexical words show varying degrees of polysemy (Durkin and Manning, 1989; Haber and Poesio, 2024; Zipf, 1945), and metaphors are pervasive even in scientific and formal domains (Steen et al., 2010a).

This work-in-progress contribution presents a novel dataset that aims to capture the regularization of metaphor with 10 polysemy patterns of English. Its intended use is to support linguistic and cognitive analyses of regular metaphor and polysemy using computational linguistic methods. In particular, we envisage experimentation with neural language models (LMs) including distributional semantic modeling, contextualized representation analysis, clustering, sense induction methods and other types of exploration and probing.

Polysemy and figurativeness cover phenomena that evade attempts to represent them in exclusively discreet or exclusively continuous ways. Recent work on polysemy shows how lexical meaning is continuous in sense individuation, sense relatedness, productivity and regularity of polysemy patterns. At the same time, grouping these continuous phenomena into symbolic, discrete-like categories can serve as a meaningful generalization when observing human language processing and building theories of language (Boleda, 2025; Futrell and Mahowald, 2025; Li, 2024; Trott and Bergen, 2023).

In this light, neural language models can serve as a suitable tool to approach such heterogeneous, fluid language phenomena. They can flexibly offer both symbolic and distributed (continuous) representations, as argued in recent position papers by Boleda (2025), Futrell and Mahowald (2025), and Grindrod (2024) and demonstrated specifically on polysemy material by Li and Armstrong (2024) and Li and Joanisse (2021). Research into linguistic capabilities of LMs shows that they are able to

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 202–211

sufficiently encode relevant linguistics information (Garí Soler and Apidianaki, 2021; Li and Joanisse, 2021; Wiedemann et al., 2019). The architectural design and the emergent linguistic capabilities of LMs make them a suitable experimental ground to study language in its complexity and heterogeneity. Our rich and systematic language resource provides the means to do so.

2. Background and Related Work

Applying the framework of regular polysemy to metaphor is not supported by many linguistic approaches to polysemy. The term was first introduced and defined by Apresjan (1974, p. 16): "Polysemy of the word A with the meanings a_i and a_j is called regular if [...] there exists at least one other word B with the meanings b_i and b_j, which are semantically distinguished from each other in exactly the same way as a_i and a_j [...]". Apresjan (1974) underlines that the typical sense extension device for this polysemy type is metonymy, while metaphorization is a source of irregular senses. Unlike metaphor that maps two disjuncted domains, metonymy operates on domain contiguity. The studies that cite and build upon Apresjan's work (1974) adopt this idea. Pustejovsky's Generative Lexicon tradition (Pustejovsky, 1991) and computational linguistic work relying on it exclude metaphor from regular polysemy investigation as well.

In contrast, a number of works in lexicography (Nimb and Pedersen, 2000; Polguère, 2018, 2022), lexicology (Berri, 2020), cognitive and psycholinguistics (Lombard et al., 2023, 2024; Srinivasan and Rabagliati, 2015), and intersecting computational domains (Barque and Chaumartin, 2009; Freihat et al., 2013; Peters and Peters, 2000) treat patterns of metaphor as regular, on par with metonymies. In particular, the psycholinguistic studies of Lombard et al. (2023) and Lombard et al. (2024) develop metrics of polysemy pattern regularity based on data from Wordnet (for English) and expert lexicological expertise (for French) showing that patterns of metaphor can be as regular as those of metonymy, with varying regularity degrees.

The concept of regular polysemy is important in the ongoing discussion on meaning representation in the human mind. Recently, hybrid approaches dominate, combining the sense enumeration approach and one meaning representation (see Haber and Poesio, 2024 for an overview). Li (2024) develops a symbolic-continuous polysemy theory based on previous experiments with neural language models (Li and Armstrong, 2024; Li and Joanisse, 2021), where regular polysemy patterns create entrenched meaning modulations that are continuous in sense individuation but allow grouping into emerging discrete-like sense clusters. Psycholinguistic evidence delivers similar accounts (Haber and Poesio, 2020; Rabagliati and Snedeker, 2013). Involving regular metaphors into the experimentation of this kind could give us a more complete picture of how patterns of meaning form. Unfortunately, the exclusion of metaphor from regular polysemy theories, and consequently from NLP experimentation that adopts these theories (if any at all), is partly a reason for the absence of a sizable corpus that could help to systematically compare two of the most important regular sense extension devices (metonymy and metaphor) on a scale. The present work contributes to the field by developing such a resource.

3. Dataset Design and Methodology

Scope. We treat conventional metaphor mappings that form patterns across the vocabulary, as per the above definition of regular polysemy (Apresjan, 1974). Conventionalized metaphors usually appear as senses of polysemous words in the dictionaries (Egg and Kordoni, 2022; Reijnierse et al., 2018; Steen et al., 2010a). If we can observe the same mappings in the sense structures in a series of words, we can consider it a regular pattern.

Source of patterns, words and senses. We borrow metaphorical patterns and their lexical realizations from previous research in regular polysemy, in particular Lombard et al. (2024). The authors examine the distribution of 15 polysemy patterns in the English lexicon by extracting lexical material from WordNet (Fellbaum, 1998; Miller, 1995), manually validating it and applying quantitative measures to assess the degree of regularity of the patterns. These measures are based on the number of words that fit the pattern, the number of words that could potentially do so and the frequency of target words in a corpus. From 15 patterns, we discarded those that were extremely irregular and would not warrant sufficient material for experimentation (less than 10 words identified as fitting the pattern), as well as those that involve highly specialized, terminological senses usually unknown to general language users. The remaining 10 patterns still fall under a wide range of regularity degrees, with some of them approached previously in psycholinguistics (Srinivasan and Rabagliati, 2015) and computational lexicology (Barque and Chaumartin, 2009).

The patterns covered at the current stage of the project are ANIMAL CRY - COMMUNICATION and NATURAL PHENOMENON - WAY OF OCCURRING. We examine both noun and verb realizations when possible. The sense mappings covered by these patterns are exemplified in Table 1 for words *roar* and *deluge*.[1] Dictionaries are used to verify that sense descriptions of selected words exhibit the intended

[1]The complete inventory of words and senses under

Pattern	Word	PoS	Literal	Metaphoric
ANIMAL CRY - COMMUNICATION	roar	V	(of a lion or other large wild animal) utter a full, deep, prolonged cry	(of a person or crowd) utter a loud, deep, prolonged sound, typically from anger, pain, or excitement; utter or express in a loud tone; (of a crowd) encourage (someone) to do something by loud shouts or cheering; laugh loudly
ANIMAL CRY - COMMUNICATION	roar	N	a full, deep, prolonged cry uttered by a lion or other large wild animal	a loud, deep sound uttered by a person or crowd, generally as an expression of pain, anger, or approval; a loud outburst of laughter
NATURAL PHENOMENON - WAY OF OCCURRING	deluge	V	overwhelm with a flood	to overwhelm with a large number or amount
NATURAL PHENOMENON - WAY OF OCCURRING	deluge	N	a severe flood; a heavy fall of rain	a great quantity of something arriving at the same time

Table 1: Sample of sense inventory. The table presents senses (*Literal* and *Metaphoric*) covered by two exemplified *Patterns* realized by words *roar* and *deluge*. Sense descriptions derived from OED.

Pattern	Word	PoS	S.	Sentence
ANIMAL CRY - COMMUNICATION	roar	V	L	The lion has **roared**: who will not fear?
ANIMAL CRY - COMMUNICATION	roar	V	M	He **roared** a great shout of laughter.
ANIMAL CRY - COMMUNICATION	roar	N	L	We ate in the London Zoo and our meals were made interesting by the chatter of monkeys and the **roar** of lions in the background.
ANIMAL CRY - COMMUNICATION	roar	N	M	Against the **roar** of rage he waved for silence.
ANIMAL CRY - COMMUNICATION	bleat	V	L	The sheep were pushing into the fold, stumbling and **bleating**.
ANIMAL CRY - COMMUNICATION	bleat	V	M	He **bleats** that his critics are talking the country down
ANIMAL CRY - COMMUNICATION	bleat	N	L	The only sounds to be heard were the sheep's teeth tearing grass and their low, rumbling **bleats**.
ANIMAL CRY - COMMUNICATION	bleat	N	M	They're trying to buy me off in hopes I'll bow to their idiotic new arrangements without a **bleat**.
NATURAL PHENOMENON - WAY OF OCCURRING	torrent	N	L	This produced a **torrent** of words but little effective action.
NATURAL PHENOMENON - WAY OF OCCURRING	torrent	N	M	A quiet river on a summer's day may be a raging **torrent** in February.
NATURAL PHENOMENON - WAY OF OCCURRING	deluge	V	L	The lake of liquid peat had burst through into the workings beneath it and was **deluging** into the colliery.
NATURAL PHENOMENON - WAY OF OCCURRING	deluge	V	M	Awards were **deluged** on him, as were titles, praise and eulogies in the national press.
NATURAL PHENOMENON - WAY OF OCCURRING	deluge	N	L	One **deluge** won't end the drought.
NATURAL PHENOMENON - WAY OF OCCURRING	deluge	N	M	Afterwards viewers sent a **deluge** of complaints about how boring the show was and asking them never to return to the Hacienda.

Table 2: Dataset sample. Column *Pattern* contains the names of regular poylysemy patterns; *Word* contains the words instantiating the pattern; *PoS* marks either a verb (*V*) or a noun (*N*); *S.* (sense) labels literal (*L*) and metaphorical (*M*) senses; *Sentence* contains extracted sentences.

metahporical pattern (Oxford English Dictionary — OED and American Heritage Dictionary — AHD)[2]. Reimann and Scheffler (2024) experimentally show that using sense descriptions from dictionaries is more reliable than human scoring on metaphor conventionalization. Such subjective annotation correlates with word frequency (Do Dinh et al., 2018; Momen et al., 2026) because human annotators tend to assign higher novelty scores to rare words. As a result, each individual annotator's exposure to less frequent vocabulary mingles with assessment of metaphor novelty. Less frequent words, however, can have perfectly conventionalized metaphoric senses listed in the dictionaries, while relatively common words may be used creatively. Our dictionary-based approach, even though time and effort-consuming, helps to avoid this confound.

Source of sentences. We present metaphorically motivated polysemes in their full sentence context to observe their naturalistic use in texts.

these patterns, as well as the list of patterns that remain to be annotated can be consulted in Appendices A and B.

[2]OED: oed.com, AHD: ahdictionary.com

Linguistic experiments often use handcrafted sentences that are tailored to the narrow purposes of each study (e.g. Cardillo et al., 2010, 2017; Lai et al., 2009). While offering a general high quality and controlled conditions of a manually created resource, these can potentially introduce bias. Momen et al. (2026) evaluate a series of language models on various pre-existing conventional metaphor datasets and find that results from synthetic data differ from the extracted data, as manually created sentences are intentional in sense distinction by lexical means. With this in mind, we opted for sentence extraction from a general language corpus (British National Corpus — BNC) (BNC Consortium, 2007) that represents naturalistic language use as opposed to synthetic data. In cases where we cannot draw enough material from BNC, we revert to Wikipedia dump [3] and Web corpora (Goldhahn et al., 2012).

Size. We selected 10 regular metaphor patterns and 10 word instantiations per pattern. As we aim for the explorations with LLMs, we draw at least 50 sentences for each word sense (the base sense and the derived metaphorical sense), so we have enough material for sensible experimentation and evaluation (see Table 2 for a sample of our dataset). To study regular metonymy and regular metaphor in comparison, we structure our dataset in a way that makes it compatible with the regular metonymy dataset by Alonso et al. (2013) and its version expanded and adapted by Li and Armstrong (2024).

4. Annotation

Sense annotation. We annotate the extracted sentences to identify the senses in which the target words are used (literal base sense and derived metaphorical sense of the pattern). Suitable sentences unambiguously evoke the sense of interest, without needing to revert to any wider context. Vague sentences that do not fulfill this condition are discarded. We plan to engage crowdsourced annotators to validate the initial sense annotation delivered by a language professional.

Frequency annotation. We sense-annotate all extracted sentences, whether or not they are suitable to be included in the dataset, in order to collect statistics on sense frequency. Usually, word frequency is controlled for in the experimentation with conventional metaphors, but, for the reasons explained above, it may not be an optimal indicator. In addition, due the unnatural, morphologically unmotivated tokenization of infrequent words, neural language models too show different performance on metaphors with low and high word frequency (Neidlein et al., 2020).

[3] dumps.wikimedia.org

A corpus-derived metric of sense frequency, on the other hand, can open interesting experimental venues by showing in what sense the word is used more often — literal or metaphoric. Klepousniotou et al. (2008) show that very conventionalized metaphoric senses become more dominant than the literal senses. Corpus-derived sense dominance can potentially be a good predictor of metaphor conventionalization, the relation that we aim to investigate in the future work. Unfortunately, existing resources on word sense frequency are scarce and have a small coverage (e.g. Bennett et al., 2016; Liu et al., 2024). One of our contributions will be an estimate of the sense frequency of each target word in the scope of the patterns and as distributed in BNC.

Annotation difficulties. For highly polysemous words with closely related, overlapping senses, identifying the base sense of a metaphoric extension can be challenging. We generally relied on the standard Metaphor Identification Procedure (Pragglejaz Group, 2007), but more extensive lexicological analyses were often needed. When complexity of the sense structure and the degree of word's polysemy are high, metaphoric sense annotation becomes a very time-consuming task.

Unlike for metonymy, sense annotation for metaphor is not complicated by underspecification, when several senses can be active at the same time (Alonso et al., 2013; Frisson, 2009). Metonymy operates on conceptual domain contiguity, but metaphor implies domain disjunction: although senses are related, they are conceptually far from each other and easier to disambiguate.

5. Discussion

The sense annotation procedure implies assigning a discrete, individual sense to a word in context. In practice, however, we had to deal with closely related, overlapping senses that were not easy to delineate. As can be seen with the example of *roar* in Table 1, such senses have been grouped together based on their similarity. The case of metaphor, in particular, brings another difficulty with it: varying underlying motivations for sense extension are grouped in the same sense category. For instance, the metaphorically motivated sense of the word *ripple* describes a manner in which events occur or unfold based on the similarity with disturbed water. It some cases, it maps dynamic visual imagery on another image (*a thing resembling a ripple or ripples in appearance or movement*), but in other, it connects it to sound (*a gentle rising and falling sound that spreads through a group of people*) and psychological states (*a particular feeling or effect that spreads through someone or something*) (sense descriptions from OED).

Consequently, even though the resulting annotation suggests discrete, uniform meaning classes, there is variation and continuity within them. This continuity can be encoded in language model representations and revealed with probing experiments by observing the internal activations in their relation to the variability of meanings in metaphorical patterns. Despite recent linguistic arguments for continuity in polysemy, the standard NLP experiments rather align with the earliest view of mental lexicon organization — sense enumeration approach (Katz and Fodor, 1963) — and highly discrete, often binary categories (polysemous vs. monosemous, regular vs. irregular, figurative vs. literal). Recent contextual language models offer the means to change this paradigm.

6. Conclusions and Future Work

This work-in-progress contribution presents a novel dataset framing metaphor as a regular sense extension of a polysemous word. Although rarely approached from this perspective, regular metaphor can contribute to the theoretical, cognitive and computational linguistic investigation of the lexical meaning structure and representation, as well as conventionalization of figurative speech. We also contribute with data on sense frequency which is of particular importance for polysemous words but largely unavailable.

We plan to use the dataset in the experimentation with neural language models. The distributional, continuous nature of these systems makes them a useful device to study polysemy and figurativeness in their complexity and heterogeneity. We continue the work on the dataset aiming to cover 10 metaphoric polysemy patterns with expert linguistic labeling, as well as subsequent crowd-sourced annotation for additional validation.

7. Limitations

Several aspects of dataset design and annotation can potentially introduce bias into the data. Manually picked sentences that very clearly realize the target sense do not faithfully represent real language use where ambiguity and vagueness are ubiquitous. This might limit our capacity to generalize insights derived from our data to language in general. A similar concern is related to the words that have been discarded from the analysis because of the difficulty to identify their base sense, or cases where metonymy and metaphor are intertwined. Discarding uncomfortable cases that do not fit into a theoretical framework distorts the picture by simplifying the complex and diverse linguistic landscape we are dealing with.

Acknowledgments

This work was funded by MCIU/AEI/10.13039/501100011033 (grants with references PID2021-128811OA-I00, PRE2022-102762, CNS2024-154902, PID2024-161928OB-I00, and AIA2025-163322-C62), and by the Galician Government (ERDF 2024-2027: Call ED431G 2023/04, and ED431B 2025/16).

8. Bibliographical References

Héctor Martínez Alonso, Bolette Sandford Pedersen, and Núria Bel. 2013. Annotation of regular polysemy and underspecification. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 725–730.

Jurij D. Apresjan. 1974. Regular polysemy. *Linguistics*, 12(142):5–32.

Lucie Barque and François-Régis Chaumartin. 2009. Regular polysemy in WordNet. *Journal for language technology and computational linguistics*, 24(2):5–18.

Marina Berri. 2020. Polisemia regular por metáfora. *Cuad. lingüíst. hisp.*, (35):19–36.

Gemma Boleda. 2025. LLMs as a synthesis between symbolic and distributed approaches to language. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 9365–9379, Suzhou, China. Association for Computational Linguistics.

Eileen R Cardillo, Christine E Watson, Gwenda L Schmidt, Alexander Kranjec, and Anjan Chatterjee. 2012. From novel to familiar: tuning the brain for metaphors. *Neuroimage*, 59(4):3212–3221.

Erik-Lân Do Dinh, Hannah Wieland, and Iryna Gurevych. 2018. Weeding out conventionalized metaphors: A corpus of novel metaphor annotations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1424, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kevin Durkin and Jocelyn Manning. 1989. Polysemy and the subjective lexicon: Semantic relatedness and the salience of intraword senses. *Journal of Psycholinguistic Research*, 18:577–612.

Markus Egg and Valia Kordoni. 2022. Metaphor annotation for German. In *Proceedings of the Thirteenth Language Resources and Evaluation Con-*

ference, pages 2556–2562, Marseille, France. European Language Resources Association.

Abed Alhakim Freihat, Fausto Giunchiglia, and Biswanath Dutta. 2013. Regular polysemy in wordnet and pattern based approach. *International Journal On Advances in Intelligent Systems*, 6:199–212.

Steven Frisson. 2009. Semantic underspecification in language processing. *Language and Linguistics Compass*, 3(1):111–127.

Richard Futrell and Kyle Mahowald. 2025. How linguistics learned to stop worrying and love the language models. *arXiv [cs.CL]*.

Aina Garí Soler and Marianna Apidianaki. 2021. Let's play mono-poly: BERT can reveal words' polysemy level and partitionability into senses. *Transactions of the Association for Computational Linguistics*, 9:825–844.

Joseph Grady, Todd Oakley, and Seana Coulson. 1999. *Blending and Metaphor*, Metaphor in Cognitive Linguistics, pages 101–124. John Benjamins Publishing Company, New York, Chichester. ISBN 9789027236814.

Jumbly Grindrod. 2024. Transformers, contextualism, and polysemy. *arXiv [cs.CL]*.

Janosch Haber and Massimo Poesio. 2020. Assessing polyseme sense similarity through co-predication acceptability and contextualised embedding distance. In *Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics*, pages 114–124.

Janosch Haber and Massimo Poesio. 2024. Polysemy — evidence from linguistics, behavioral science, and contextualized language models. *Comput. Linguist. Assoc. Comput. Linguist.*, 50(1):1–67.

Jerrold Katz and Jerry Fodor. 1963. The structure of a semantic theory. *Language*, 39:170–210.

Ekaterini Klepousniotou, Debra Titone, and Carolina Romero. 2008. Making sense of word senses: the comprehension of polysemy depends on sense overlap. *J. Exp. Psychol. Learn. Mem. Cogn.*, 34(6):1534–1543.

Vicky Tzuyin Lai, Tim Curran, and Lise Menn. 2009. Comprehending conventional and novel metaphors: an ERP study. *Brain Res.*, 1284:145–155.

George Lakoff and Mark Johnson. 1980. *Metaphors We Live By*. University of Chicago Press, Chicago, IL.

Ronald W Langacker. 2017. Entrenchment in cognitive grammar. In *Entrenchment and the psychology of language learning: How we reorganize and adapt linguistic knowledge*, pages 39–56. American Psychological Association, Washington.

Jiangtian Li. 2024. Semantic minimalism and the continuous nature of polysemy. *Mind Lang.*

Jiangtian Li and Blair C Armstrong. 2024. Probing the representational structure of regular polysemy via sense analogy questions: Insights from contextual word vectors. *Cogn. Sci.*, 48(3):e13416.

Jiangtian Li and Marc F Joanisse. 2021. Word senses as clusters of meaning modulations: A computational model of polysemy. *Cogn. Sci.*, 45(4):e12955.

Alizée Lombard, Richard Huyghe, Lucie Barque, and Doriane Gras. 2023. Regular polysemy and novel word-sense identification. *The Mental Lexicon*, 18(1):94–119.

Alizée Lombard, Anastasia Ulicheva, Maria Korochkina, and Kathy Rastle. 2024. The regularity of polysemy patterns in the mind: Computational and experimental data. *Glossa Psycholinguistics*, 3(1).

Omar Momen, Emilie Sitter, Berenike Herrmann, and Sina Zarrieß. 2026. Surprisal and metaphor novelty: Moderate correlations and divergent scaling effects. *arXiv [cs.CL]*.

Arthur Neidlein, Philip Wiesenbach, and Katja Markert. 2020. An analysis of language models for metaphor recognition. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3722–3736, Stroudsburg, PA, USA. International Committee on Computational Linguistics.

Sanni Nimb and Bolette Sanford Pedersen. 2000. Treating metaphoric senses in a danish computational lexicon –different cases of regular polysemy. In *Proceedings of the 9th EURALEX International Congress*, pages 679–691, Stuttgart, Germany. Institut für Maschinelle Sprachverarbeitung.

Wim Peters and Ivonne Peters. 2000. Lexicalised systematic polysemy in WordNet. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'00)*, Athens, Greece. European Language Resources Association (ELRA).

Alain Polguère. 2018. A lexicographic approach to the study of copolysemy relations. *Russ. J. Linguist.*, 22(4):788–820.

Alain Polguère. 2022. Fishing out regular polysemy with the gener (generic) relation. *Lexique*, (31).

Pragglejaz Group. 2007. MIP: A method for identifying metaphorically used words in discourse. *Metaphor Symb.*, 22(1):1–39.

James Pustejovsky. 1991. The Generative Lexicon. *Computational Linguistics*, 17(4):409–441.

Hugh Rabagliati and Jesse Snedeker. 2013. The truth about chickens and bats: ambiguity avoidance distinguishes types of polysemy: Ambiguity avoidance distinguishes types of polysemy. *Psychol. Sci.*, 24(7):1354–1360.

W Gudrun Reijnierse, Christian Burgers, Tina Krennmayr, and Gerard J Steen. 2018. DMIP: A method for identifying potentially deliberate metaphor in language use. *Corpus Pragmat.*, 2(2):129–147.

Sebastian Reimann and Tatjana Scheffler. 2024. When is a metaphor actually novel? annotating metaphor novelty in the context of automatic metaphor detection. In *Proceedings of the 18th Linguistic Annotation Workshop (LAW-XVIII)*, pages 87–97, St. Julians, Malta. Association for Computational Linguistics.

Mahesh Srinivasan and Hugh Rabagliati. 2015. How concepts and conventions structure the lexicon: Cross-linguistic evidence from polysemy. *Lingua*, 157:124–152.

Gerard J Steen, Aletta G Dorst, J Berenike Herrmann, Anna A Kaal, and Tina Krennmayr. 2010a. Metaphor in usage. *Cogn. Linguist.*, 21(4):765–796.

G.J. Steen, A.G. Dorst, J.B. Herrmann, A.A. Kaal, T. Krennmayr, and T. Pasma. 2010b. *A method for linguistic metaphor identification. From MIP to MIPVU.* Number 14 in Converging Evidence in Language and Communication Research. John Benjamins.

Sean Trott and Benjamin Bergen. 2023. Word meaning is both categorical and continuous. *Psychol. Rev.*, 130(5):1239–1261.

Gregor Wiedemann, Steffen Remus, Avi Chawla, and Chris Biemann. 2019. Does bert make any sense? interpretable word sense disambiguation with contextualized embeddings. In *Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019)*.

George Kingsley Zipf. 1945. The meaning-frequency relationship of words. *The Journal of general psychology*, 33:251–6.

9. Language Resource References

Bennett, Andrew and Baldwin, Timothy and Lau, Jey Han and McCarthy, Diana and Bond, Francis. 2016. *LexSemTm: A Semantic Dataset Based on All-words Unsupervised Sense Distribution Learning*. Association for Computational Linguistics. PID https://aclanthology.org/P16-1143/.

BNC Consortium. 2007. *The British National Corpus, version 3 (BNC XML Edition)*. Oxford Text Archive. PID http://hdl.handle.net/20.500.14106/2554.

Cardillo, Eileen R and Schmidt, Gwenda L and Kranjec, Alexander and Chatterjee, Anjan. 2010. *Stimulus design is an obstacle course: 560 matched literal and metaphorical sentences for testing neural hypotheses about metaphor*. Springer Science and Business Media LLC. PID https://doi.org/10.3758/BRM.42.3.651.

Cardillo, Eileen R and Watson, Christine and Chatterjee, Anjan. 2017. *Stimulus needs are a moving target: 240 additional matched literal and metaphorical sentences for testing neural hypotheses about metaphor*. Springer. PID https://doi.org/10.3758/s13428-016-0717-1.

Fellbaum, Christiane. 1998. *WordNet: An Electronic Lexical Database*. The MIT Press. PID https://doi.org/10.7551/mitpress/7287.001.0001.

Goldhahn, Dirk and Eckart, Thomas and Quasthoff, Uwe. 2012. *Building Large Monolingual Dictionaries at the Leipzig Corpora Collection: From 100 to 200 Languages*. European Language Resources Association (ELRA). PID http://www.lrec-conf.org/proceedings/lrec2012/pdf/327_Paper.pdf.

Liu, Lei and Gong, Tongxi and Shi, Jianjun and Guo, Yi. 2024. *A high-frequency sense list*. Frontiers Media SA. PID https://doi.org/10.3389/fpsyg.2024.1430060.

Miller, George A. 1995. *WordNet: a lexical database for English*. Association for Computing Machinery. PID https://doi.org/10.1145/219717.219748.

Appendices

A. Pattern List

Below we list polysemy patterns covered by the present study.

1. ANIMAL - ARTIFACT
2. ANIMAL - PERSON
3. ANIMAL CRY - COMMUNICATION
4. ARTIFACT - MESSAGE
5. BODY PART - OBJECT PART
6. NATURAL EVENT - WAY OF OCCURRING
7. NATURAL EVENT - SOCIAL EVENT
8. PERSON - ANIMAL
9. PERSON - ARTIFACT
10. PHYSICAL PROPERTY - PSYCHOLOGICAL PROPERTY

B. Sense Inventories

Two of the patterns listed in the Appendix A have been annotated: ANIMAL CRY - COMMUNICATION and NATURAL PHENOMENON - WAY OF OCCURRING. Tables 3 and 4 present sense inventories for these patterns as covered by the annotated words. The sense descriptions are mainly derived from the OED. In certain cases, we draw additional material from AHD:

- when OED offers a circular definition that reuses the term being defined (or a morphologically related word) in the meaning description. For instance, the metaphorical sense of the verb *shower* is described by OED using its nominal counterpart:

  > *(of a mass of small things) fall or be thrown in a **shower**: bits of broken glass **showered** over me.*

- in a similar situation to the above, but when a synonym is used as a sense description, especially if this synonym is one of the target lexemes under study. For example, the literal sense of the verb *inundate* is simply defined as

  > *flood: the islands may be the first to be **inundated** as sea levels rise*.

- when OED merges the base literal sense with the derived metaphorical sense in a single sense description, as, for instance, in the case of the verb *bellow*:

  > *(of a person or animal) emit a deep loud roar, typically in pain or anger: he **bellowed** in agony.*

- when AHD lists more pattern-related senses than OED does, which is also reflected in the corpus annotation. This is the case for such words as *howl*, *riffle* and *torrent*.

Word	PoS	Literal	Metaphorical	Source
bark	V	(of a dog, fox, or seal) give a bark	(of a person) make a sound resembling a bark; utter (a command or question) abruptly or aggressively; call out in order to sell or advertise something	OED
	N	the sharp explosive cry of a dog, fox, or seal	a sound resembling a bark, typically one made by someone laughing or coughing	OED
bellow	V	To make the deep roaring sound characteristic of a bull.	sing (a song) loudly and tunelessly; shout something with a deep loud roar	OED
	N	The roar of a large animal, such as a bull.	A very loud utterance or other sound.	AHD
bleat	V	(of a sheep, goat, or calf) make a characteristic weak, wavering cry	speak or complain in a weak, querulous, or foolish way	OED
	N	the weak, wavering cry made by a sheep, goat, or calf	a person's weak or plaintive cry; a complaint	OED
bray	V	(of a donkey or mule) utter a bray	(of a person) speak or laugh loudly and harshly	OED
	N	the loud, harsh cry of a donkey or mule	a sound, voice, or laugh resembling a bray	OED
cackle	V	(of a bird, especially a hen or goose) give a raucous clucking cry	laugh in a loud, harsh way; talk at length without acting on what is said	OED
	N	the raucous clucking cry of a bird such as a hen or a goose	a loud, harsh laugh	OED
chirp	V	(of a small bird or an insect) make a short, sharp, high-pitched sound	(of a person) say something in a lively and cheerful way	OED
	N	A short, high-pitched sound, such as the one a small bird or insect makes	To make a short, high-pitched sound.	AHD
growl	V	(of an animal, especially a dog) make a low guttural sound in the throat	(of a person) say something in a low grating voice, typically in a hostile or angry	OED
	N	a low guttural sound made in the throat by a hostile dog or other animal	low guttural sound or utterance made by a person, especially to express hostility or anger	OED
grunt	V	(of an animal, especially a pig) make a low, short guttural sound	(of a person) make a low inarticulate sound, typically to express effort or indicate assent	OED
	N	a low, short guttural sound made by an animal or a person	a low, short guttural sound made by an animal or a person	OED
howl	V	make a howling sound	weep and cry out loudly; to laugh heartily	OED+AHD
	N	a long, doleful cry uttered by an animal such as a dog or wolf.	a loud cry of pain, fear, anger, or amusement;	OED
roar	V	(of a lion or other large wild animal) utter a full, deep, prolonged cry	(of a person or crowd) utter a loud, deep, prolonged sound, typically from anger, pain, or excitement; utter or express in a loud tone; (of a crowd) encourage (someone) to do something by loud shouts or cheering; laugh loudly	OED
	N	a full, deep, prolonged cry uttered by a lion or other large wild animal	a loud, deep sound uttered by a person or crowd, generally as an expression of pain, anger, or approval; a loud outburst of laughter	OED

Table 3: Sense inventory for the pattern ANIMAL CRY - COMMUNICATION. The table presents senses (*Literal* and *Metaphoric*) covered by each *Word*, as well as the *Source* of the definitions. *PoS* marks either a verb (*V*) or a noun (*N*).

Word	PoS	Literal	Metaphoric	Source
backwash	N	the motion of receding waves; a backward current created by an object moving through water or air	the unpleasant after-effects of an event	OED
deluge	V	overwhelm with a flood	to overwhelm with a large number or amount	OED
	N	a severe flood; a heavy fall of rain	a great quantity of something arriving at the same time	OED
flood	V	cover or submerge (an area) with water in a flood; become covered or submerged by a flood; become swollen and overflow (its banks)	arrive in overwhelming amounts or quantities; completely fill or suffuse; overwhelm with large amounts or quantities	OED
	N	an overflow of a large amount of water beyond its normal limits, especially over what is normally dry land; the inflow of the tide	an outpouring of tears; an overwhelming quantity of things or people happening or appearing at the same time	OED
inundation	N	flooding	an overwhelming abundance of people or things	OED
inundate	V	to cover with water, especially floodwaters.	overwhelm (someone) with things or people to be dealt with	OED+AHD
rain	V	rain falls; (of the sky, the clouds, etc.) send down rain	fall or cause to fall in large or overwhelming quantities;	OED
	N	the condensed moisture of the atmosphere falling visibly in separate drops; falls of rain	a large or overwhelming quantity of things that fall or descend	OED
riffle	V	to flow in rough waves or become choppy, as water.	to thumb through (the pages of a book, for example); to shuffle cards;	OED
	N	a rocky or shallow part of a stream or river where the water flows brokenly; a stretch of choppy water caused by such a shoal or sandbar; a rapid; a wave or ripple in such water.	an act or sound of riffling through something; The act or an instance of shuffling cards; disturb the surface of; ruffle:	OED+AHD
ripple	V	(of water) form or flow with a series of small waves on the surface; cause (the surface of water) to form small waves	move in a way resembling a series of small waves; (of a sound or feeling) spread through a person, group, or place	OED
	N	a small wave or series of waves on the surface of water, especially as caused by a slight breeze or an object dropping into it;	a thing resembling a ripple or ripples in appearance or movement; a gentle rising and falling sound that spreads through a group of people; a particular feeling or effect that spreads through someone or something	OED
shower	V	-	to throw or cause small things or pieces to fall over; throw (a number of things) all at once towards someone; give someone a great number of (things); give a great number of things to (someone)	OED+AHD
	N	a brief and usually light fall of rain, hail, sleet, or snow	a mass of small things falling or moving at the same time; a large number of things happening or given at the same time	OED
torrent	N	a strong and fast-moving stream of water or other liquid; a heavy downpour; a deluge.	an overwhelming outpouring of (something, typically words)	OED+AHD
wave	V	-	move to and from with a swaying motion while remaining fixed to one point	OED
	N	a long body of water curling into an arched form and breaking on the shore; a ridge of water between two depressions in open water	a shape regarded as resembling a breaking wave; a sudden occurrence of or increase in a phenomenon, feeling, or emotion	OED
roar	V	(of a lion or other large wild animal) utter a full, deep, prolonged cry	(of a person or crowd) utter a loud, deep, prolonged sound, typically from anger, pain, or excitement; utter or express in a loud tone; (of a crowd) encourage (someone) to do something by loud shouts or cheering; laugh loudly	OED
roar	N	a full, deep, prolonged cry uttered by a lion or other large wild animal	a loud, deep sound uttered by a person or crowd, generally as an expression of pain, anger, or approval; a loud outburst of laughter	OED

Table 4: Sense inventory for the pattern NATURAL PHENOMENON - WAY OF OCCURRING. The table presents senses (*Literal* and *Metaphoric*) covered by each *Word*, as well as the *Source* of the definitions. *PoS* marks either a verb (*V*) or a noun (*N*).

Domain-Specific Considerations in the Preparation of Specialized Corpora: A Case Study on a Corpus of German Sermons

Cora Haiber[1,2,3], **Adam Roussel**[1], **Stefanie Dipper**[1]

[1]Ruhr-University Bochum
Universitätsstraße 150
44801 Bochum
Germany
firstname.lastname@rub.de

[2]Saarland University
Campus
66123 Saarbrücken
Germany

[3]Zuse School ELIZA
Hochschulstraße 10
64289 Darmstadt
Germany

Abstract

We present a new corpus of contemporary German sermons and describe the steps taken in its preparation. We apply a semi-automatic approach to sentence segmentation, tokenization, and lemmatization, utilizing annotation guidelines that are specialized to this domain. In the process of preparing these data, we find that state-of-the-art tools for these tasks still make problematic errors, especially with non-standard data, despite apparently very high performance on common benchmarks. We obtain test scores of F_1 = 96.69 % for sentence segmentation, F_1 = 99.99 % for tokenization, and acc = 64.00 % for lemmatization with our domain-adapted models and show that domain-adaptation improves performance over state-of-the-art models for the token and sentence segmentation tasks.

Keywords: corpus preparation, language resources, tokenization, segmentation, lemmatization, semi-automatic annotation

1. Introduction

In this paper we introduce a new specialized corpus of contemporary German sermons, which currently includes a range of metadata specific to the domain, as well as semi-automatically annotated tokenization, sentence boundaries, and lemmatization. Further levels of linguistic annotations are planned, including POS, morphological features, and eventually dependency parses.

In the following, we will outline the process of preparing this corpus and laying the foundations for the annotations to come. The benchmark evaluation of these initial tasks of tokenization, sentence segmentation, and lemmatization usually results in rather high accuracy and F_1 scores, such that one might reasonably have the impression that these are solved tasks, and automatic tools for this purpose can be applied directly with few reservations.

However, when it comes to the preparation of corpora, the comparatively few errors that there are have disproportionate consequences, since tokenization, sentence segmentation, and lemmatization are tasks that are located towards the beginning of several corpus preparation pipelines. Any errors in these early steps will propagate through all subsequent layers of annotation. Furthermore, when it comes to domain-specific corpora and non-standard text types, these errors are not necessarily so few in number, as the requirements of these domains may differ significantly from the newspaper articles that still form the backbone of respective off-the-shelf models.

Thus some of the problems that we have encountered during the process of adding tokenization, sentence segmentation, and lemmatization to this corpus show that there are still improvements to be made, especially as regards the adaptation of corpus preparation tools to specialized domains.

After a review of related work in Section 2, we provide a more detailed description of the corpus and its contents in Section 3. Section 4 provides an overview of the aspects of the annotation guidelines that are more particular to this specialized domain. Section 5 will then give a detailed analysis of the process of training and evaluating algorithms and models for extending our annotations according to these specialized guidelines to the rest of the corpus. Finally, in Section 6, we will discuss some of the consequences and implications of our findings for future work and conclude. Our data, guidelines, scripts, and models are available in a GitLab repository[1].

2. Related Work

2.1. Tokenization and Sentence Segmentation

Other specialized corpora are faced with similar kinds of tokenization and segmentation problems. For instance, Dipper and Laarmann-Quante (2024) discuss some of the issues that they found during the preparation of a parsed corpus of poetry, specifically with regard to the implications of sentence segmentation for dependency parsing of the data.

[1]https://gitlab.ruhr-uni-bochum.de/compchist/slide-2026-specialized-corpora

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 212–223

Beißwenger et al. (2015) present guidelines for tokenization of a domain-specific corpus, namely one concerning computer-mediated communication. Their guidelines must account for a number of phenomena that are quite frequent in such data, such as URLs, hashtags, and emoticons, but not in the more standard datasets that have guided tokenization approaches traditionally, either in the form of training data or as test data for rule-based methods. Similarly to the approach we have adopted for this corpus, Beißwenger et al. (2015) employ a manual correction step, in which annotators work with a one-token-per-line plain text format.

Diewald et al. (2022) evaluate state-of-the-art tokenization tools for German in terms of adaptability for new or different corpora and show how tokenization can be specific to a certain text type. They further emphasize the importance of high-quality tokenization since tokenization errors cascade through all processing steps aligning with our interests of a sensible and reliable tokenization to act as a foundation for higher-level processing steps.

The evaluation in Ortmann et al. (2019) draws attention to the relationship between text types and the performance of off-the-shelf language processing tools. Specifically, they show how tokenization errors often correspond to conventions that are specific to a given text type.

2.2. Lemmatization

In accordance with Toporkov and Agerri (2024), we define lemmatization as producing, from a given inflected word, its lexicon form, which we call lemma. In general, lemma forms correspond to the canonical dictionary form, e.g., for verbs this is often the infinitive, for nouns the nominative singular. However, for morphologically inflected languages such as German, there are many special cases that are handled differently in different guidelines. Table 1 shows some examples from the guidelines of the two standard German corpora, TIGER (Crysmann et al., 2005) and Tüba-D/Z (Telljohann et al., 2017), together with the corresponding forms that we use in the sermons.The table shows that in TIGER, information about capitalization, which distinguishes nouns in German, is not included in the lemma.

Common modeling approaches for lemmatization can be categorized into neural, rule- or pattern-based, and lookup-based models. Lookup-based models such as the WordNet lemmatizer (Miller, 1994) rely on pre-compiled dictionaries or morphological databases and are inherently limited in coverage. Rule-based lemmatizers such as spaCy (Honnibal et al., 2020) lemmatizers apply handcrafted or automatically induced heuristics. They are highly interpretable and efficient, but may lack the flexibility to handle the complexities of morphologically rich languages such as German. Neural models such as Stanza (Qi et al., 2020) lemmatizers or Lematus (Bergmanis and Goldwater, 2018) are often the most flexible option with high coverage since they treat the task as a sequence-to-sequence model or a classification task and leverage contextual information to resolve ambiguity, but might lack reliability and linguistic interpretability (see Section 5.3.2).

Toporkov and Agerri (2024) emphasize the importance of evaluating lemmatization models on out-of-domain test data and find that models using simple UPOS tags outperform those trained with fine-grained morphological features. Dorkin and Sirts (2024) compare current approaches to lemmatization on Estonian and conclude that ensembles of different modeling techniques might be the best possible approach to the task of lemmatization as of today. Hence, we choose to expand GermaLemma (Konrad, 2019), an ensemble method for the lemmatization of part-of-speech (POS) tagged German sentences, which combines rule- and lookup based methods with a large lemma dictionary based on the TIGER corpus (Brants et al., 2004), functions from the CLiPS "Pattern" package (De Smedt and Daelemans, 2012), and an algorithm to split compounds.

3. Corpus description

The Corpus of Contemporary German-language Protestant Sermons consists of a collection of 1 068 German-language sermons published by the Zentrum für evangelische Gottesdienst- und Predigtkultur ('Center for Protestant Mass and Sermon Culture')[2] between 2011 and 2023. The sermons in our corpus are typically based on a specific Bible passage, which is explained and commented on, and some include literal citations of the respective passage. The corpus includes sermons from 109 authors who generously granted us permission to include their works, contains approximately 1.93 M tokens in total, and is available under a CC BY-NC-SA 4.0 license[3]: https://linguistics.rub.de/predigtenkorpus.

The texts are collected in their original HTML format and automatically converted to an internal JSON-based standoff format. In the process, we preserve as much information as possible from the original documents. This includes metadata pertaining to the author, the date published, and the relevant Bible passage, including a URL that points to a digital version of that Bible passage, among other things. It also includes text-structural elements, such as headings and line breaks, when these are included in the markup, and emphasized

[2] https://predigten.evangelisch.de/

[3] https://creativecommons.org/licenses/by-nc-sa/4.0/

Word class	Word forms	TIGER	Tüba-D/Z	Sermons
Articles	der / die / das 'the'	der	der / die / das	der
	ein / eine 'a'	ein	ein / eine	ein
Pronouns	manchen / manches 'some'	mancher	mancher / manches	mancher
	wer / wem / was 'who, what'	wer / wem / was	wer / wer / was	wer / wer / was
Deadj. nouns	(ein) Junger 'a youth'	junge	Junger	Junge
	Nettes '(something) nice'	nette	Nettes	Nette
Deverbal nouns	(das) Lesen 'reading'	lesen	Lesen	Lesen
	(der) Angeklagte 'the accused'	angeklagter	Angeklagter	Angeklagte
	(die/der) Vorsitzende 'the presiding (person)'	vorsitzend	Vorsitzender	Vorsitzende
Fused prep.	im 'in the'	in	in	im

Table 1: Example lemmas from the German corpora TIGER and Tüba-D/Z.

passages (using `<em>` tags), which are usually quotations from the Bible. Since the Bible passages themselves use a very different form of language than the sermons themselves, it is of particular interest for the annotation process to be able to distinguish these passages from the surrounding text, and preserving this markup makes this possible with a fair degree of reliability.

4. Annotation

All 1068 texts were initially tokenized and split into sentences using SoMaJo with its `de_CMC` ruleset (Proisl and Uhrig, 2016). Of these automatically tokenized documents, 155 randomly selected sermons were then manually corrected according to the domain-specific tokenization guidelines presented below. Later, in Section 5, these corrected data will be used in the training of segmentation and tokenization models, with which we plan to process the remaining corpus documents.

Note that the following sections don't contain the full contents of the guidelines used for the given tasks, but rather serve to give an impression of some of their finer points, i.e. the aspects that apply to the domain of sermons in particular. The full guidelines are available as part of the documentation, which can be found in the corpus repository.

4.1. Tokenization and Sentence Segmentation

In the following examples, tokens are separated by space characters, and sentences are each presented on a new line, indicated by the line numbers given at the left. Where sentences are too long to fit in a column, they are wrapped and this is indicated by ↩. Continuations of logical lines are indented to reflect that these line breaks do not correspond to sentence boundaries as usual.[4] English translations of the relevant passages are provided in italics.

Biblical citations. Citations from the Bible usually provide an abbreviation of the name of the book, a chapter number, and a verse number, i.e. `2. Kor 13,11` refers to the second letter to the Corinthians, chapter 13, verse 11. Each citation represents a complex reference to a Bible passage and should therefore be treated as a single token. One simple way to achieve this is to replace space characters (i.e., U+0020) within these reference by underscores (_, U+005F), to indicate visual spacing and also the instance's acting as a single token, as in `2._Kor_13,11`. This is similar to the format sometimes used for multi-word expressions so that they can be treated as single tokens, e.g. by `word2phrase` (Mikolov et al., 2013), the `mwetoolkit`[5] (Ramisch, 2015), and `stanza` (Qi et al., 2020).

Multiple citations should be multiple tokens. These are usually separated by punctuation marks such as the semicolon `;`. Citations and sequences of citations, when they come at the end of an otherwise complete sentence, should be considered a separate sentence:

```
1 Freuet euch !
   'Rejoice !'
2 Kap_3,1 ; 2._Kor_13,11 ; ↩
      1._Thess_5,16
```

If a citation acts as an argument, i.e. it is integrated syntactically into the surrounding text, or it is part of the title, it is not considered to be a separate sentence.

```
1 Predigt zu Lukas_17,11-19
   'Sermon on Luke_17,11-19'
```

Quoted material. Quotations (in this corpus, these are usually Biblical in origin) are segmented internally if they contain multiple sentences:

```
2
```

[4] Note that this differs from the format that was used during the annotation process, in which, for easier editing, each line contained one token, and sentence boundaries were represented by an empty line.

[5] Though note that `mwetoolkit` also supports many other formats. https://gitlab.com/mwetoolkit/mwetoolkit3

```
1 5
3 Eure Güte laßt kund sein allen ↩
     Menschen !
  'Let your kindness be known to ↩
     all people !'
4 Der Herr ist nahe !
  'The Lord is near !'
5 Tit_3,2
```

Quoted material may be preceded by introductory comments. These should be annotated as a single sentence whenever the quoted material is considered part of the previous sentence syntactically, which is usually the case.

```
1 Dem Briefpapier vertraut er an ↩
     : Paulus , berufen zum ↩
     Apostel Christi Jesu durch ↩
     ...
  'To the letterhead he entrusts ↩
     : Paul , called to be an ↩
     apostle of Christ Jesus ↩
     through ...'
```

Headers and other metadata. If the sermon contains a title, subtitle, author, etc, they are all segmented separately.

```
1 Versöhnung : Christliches ↩
     Alleinstellungsmerkmal !?
  'Reconciliation: A uniquely ↩
     Christian trait !?'
2 Hans Meyer
  'Hans Meyer'
3 Liebe Gemeinde !
  'Dear congregation !'
```

4.2. Lemmatization

Our guidelines largely follow those of TIGER. The most important differences are (see Table 1):

- Deadjectival and deverbal nouns: We follow the original upper/lower case spelling and adopt this for the lemma (see TüBa-D/Z), regardless of whether it is a lexicalized form.
- Deadjectival nouns: The lemma adopts the ending of the weakly inflected nominative singular, which is *-e* in all three genders (see TIGER). This ending is also used for nominalized participles.
- Personal pronouns: Word forms that are generally ambiguous are lemmatized with an ambiguous form: *ihm* → *er_es; ihr* → *sie_ihr*.
- Proper nouns: Unusual capitalization is normalized in the lemma, e.g. *der Heilige Geist* → *heilig; HERR* → *Herr*
- Fused prepositions: To encode the difference from normal prepositions, the lemma form corresponds to the word form: *im* → *im*.

5. Models

To scale the applicability of our guidelines to the size of the corpus, we develop models for sentence segmentation, tokenization, and lemmatization. Our overarching goals are to robustly and efficiently model the domain-specific particularities of our texts according to our guidelines and to investigate the transferability of off-the-shelf tools to a specialized domain. While we primarily deal with Biblical citations and quoted Bible passages, other particularities will emerge in other domains. What they have in common is that expert knowledge and modeling adaptations might be required to adequately process them. We ascertain that deviating from standard modeling approaches is appropriate for various domain-specific problems and that quantitative evaluation scores of state-of-the-art models are not necessarily informative of their true generalization abilities.

5.1. Segmentation

Common segmentation algorithms, such as Punkt (Kiss and Strunk, 2006), largely rely on punctuation to detect sentence boundaries, which is unsuitable for our sermons due to three fallacies. Firstly, though the texts were published in written form, many of them are more adjacent to transcripts of spoken language – manifesting in the fact that the punctuation often reflects rhetorical indications, such as pauses in speech, rather than sentence boundaries. Secondly, the texts contain numerous interjections of Biblical citations with parentheses, additional punctuation, and abbreviations of Bible chapters (e.g. *Lk.*, *Mk.*) as well as direct speech, where a punctuation mark does not necessarily reflect a sentence boundary in the matrix clause. Thirdly, the sermons frequently contain line numbers, which we choose to segment separately but which may not be separated by any punctuation mark.

5.1.1. Methods

To ensure robustness to the above particularities, we follow the approach of Frohmann et al. (2024), who present a collection of segmentation models (SaT) with less reliance on punctuation and promising experiments with Low-Rank-Adaptation (LoRA; Hu et al., 2021) for specialized domains. They train subword-based multilingual encoder language models (LMs) in a self-supervised manner on 85 languages sampled from the mC4 corpus

Split	%	#Text	#Sent	#Tok
train	70	106	12 103	208 589
val	15	23	2 668	45 287
test	15	26	2 829	47 791

Table 2: Split of manually corrected data for segmentation model development reporting percentage of data (%), number of texts (#Text), number of sentences (#Sent), and number of tokens (#Tok) per subset.

Base	F1	R	P
sat-1l	0.71	0.94	0.57
sat-3l	0.67	0.96	0.52
sat-6l	0.76	0.95	0.63
sat-9l	0.80	0.94	0.69
sat-12l	**0.81**	**0.93**	**0.71**
sat-1l-sm	0.90	0.87	0.92
sat-3l-sm	0.93	0.91	0.95
sat-6l-sm	0.93	0.94	0.93
sat-12l-sm	**0.95**	**0.94**	**0.96**

Table 3: Evaluation of pretrained SaT models on the validation set reporting base model, F1 score, recall (R), and precision (P).

(Raffel et al., 2019) and subsequently train the self-supervised models on a supervised mixture of already segmented sentences (`sm-models`). We split the 155 manually corrected sermons as presented in Table 2 and first evaluate the pretrained models on our validation set (see Table 3). Model performance is measured in terms of F1 score as in the original paper[6].

For both classes, unsupervised and supervised, we find that the largest models *sat-12l* and *sat-12l-sm* perform best and that the *sm*-versions perform better than the completely unsupervised models. We train LoRA adapters for each of the two highest-performing models on our training set, of which the size approximately compares to the 10 000 sentences that Frohmann et al. (2024) use for LoRA training.

5.1.2. Results

Matching the findings of Frohmann et al. (2024), we find that the *sm*-versions do not benefit from LoRA as much as the unsupervised models and perform slightly worse with adapters even though the base *sm*-models had significantly higher F1 scores. Furthermore, model scores only increase for a given number of epochs *k* as demonstrated in Table 4. The overall best model on the validation set consists of *sat-12l* with a LoRA head trained for 6 epochs and reaches F_1 = 96.69 % on the validation set. After epoch 6, precision improves further but at the cost of recall. A final evaluation on the test set yields F_1 = 96.86 % with 94.97 % recall and 98.82 % precision. Compared to the validation set, we obtain slightly lower recall and slightly higher precision, but overall good generalization abilities.

A qualitative analysis of the model outputs reveal that Biblical citations are still a major source of errors and are not segmented reliably.

[6] Frohmann et al. (2024) suggest character-level F1 scores for the positive labels to compare ground truth and predicted sentence boundaries. I.e. for each predicted sentence, we check whether the right sentence border is in the correct position.

Base	#Ep	F1	R	P
sat-12l	5	0.96	0.94	0.98
sat-12l	6	**0.97**	**0.95**	0.98
sat-12l	7	0.96	0.95	0.98
sat-12l	10	0.96	0.93	0.99
sat-12l	20	0.25	0.14	**0.99**
sat-12l-sm	4	0.95	0.95	0.96
sat-12l-sm	5	0.96	0.95	0.97
sat-12l-sm	6	0.96	0.94	0.98

Table 4: Evaluation of LoRA adapters with different hyperparameter configurations on the validation set, reporting base model, number of epochs trained (#Ep), F1 score, recall (R), and precision (P).

```
1 Ja, Gesetzlichkeit ist ↩
    eigentlich nichts als ↩
    Ärgernis an der " ↩
    herrlichen Freiheit der ↩
    Kinder Gottes " , wie ↩
    Paulus sagt ( Röm._8,21 ) .
  'Yes , the law is really ↩
    nothing but a hindrance to ↩
    the ‘ wonderful freedom of ↩
    the children of God ' , as ↩
    Paul says ( Röm._8,21 )'
2 Ich lese den gesamten ↩
    Textabschnitt und stelle ↩
    uns damit die Szene ↩
    bildlich vor Augen ↩
    Mi_6,1-8.
  'I read the entire passage and ↩
    use it to vividly picture ↩
    the scene in our minds ↩
    Mi_6-,18.'
3 Wir haben also heute ↩
```

```
      interessanterweise einen ↩
      alttestamentlichen Text zum ↩
      Osterfest !
   'So , interestingly enough , ↩
      today we have an Old ↩
      Testament passage about ↩
      Easter !'
4  Jesaja_25,8-9
5  Es ist Matthäus , der sie ↩
      schildert : " Und Petrus ↩
      ging hinaus und weinte ↩
      bitterlich . " .
   'It is Matthew who describes ↩
      it : "And Peter went out ↩
      and wept bitterly. "'
6  ( Mt_26,75 )
```

According to the guidelines, each of the examples above should be segmented into two sentences, but the segmentation model only splits the sentences in 3–4 and 5–6 correctly but not in 1 and 2. There is no clear tendency as to which cases are segmented incorrectly other than punctuation: Citations are usually segmented correctly when there is no subsequent punctuation mark and usually segmented incorrectly when there is.

5.2. Tokenization

Sensible tokens are fundamental to semantically motivated tasks such as metaphor detection, where annotators as well as classifiers rely on a reliable lexicon of candidate words. The assumption that tokenization is a solved task based on very high quantitative evaluation metrics is challenged by the intricacies of domain-specific text.

5.2.1. Methods

We apply the neural tokenizer model of Qi et al. (2018), who propose the combination of a bidirectional LSTM and a 1-layer CNN, and provide a training interface for custom models within the Stanza framework (Qi et al., 2020). The default model for German is trained on a dataset sampled from the GSD (Nivre et al., 2016) and the HDT (Borges Völker et al., 2019) treebanks and reaches F_1 = 99.62 % on a GSD test set and F_1 = 100.00 % on a HDT test set. We evaluate the off-the-shelf model on the same validation set of sermons that was used for the development of the segmentation model and obtain a comparatively low accuracy of 95.83 %, even when citations from the Bible are already joined by underscores. A qualitative analysis reveals that the model tends to split these tokens specifically at underscores and other punctuation. Since the models distributed with Stanza are not inherently re-trainable, we default to training a new model with the same architecture on a mixture of the GSD training data and our training set of sermons.

5.2.2. Results

The best model with a mixing ratio of approximately 2:1 of sermons and GSD sentences illustrated in Table 6 yields F_1 = 99.99 % on the test set[7]. We evaluate qualitatively on our test set in two settings: *simple*, where Biblical citations have already been preprocessed and joined by underscores, and *hard*, where this is not the case and subtokens of the citations are split by spaces. We utilize our manually corrected sermons for this, but for future utility this preprocessing step could be done with either a rule-based script or a specialized model for named entity recognition. In the *easy* setting, the tokenizer learns to not split citations at underscores, validating our approach to adapt the default model to our domain. However in the *hard* setting, citations are split into subtokens and the model does not learn to recognize them as constructs adjacent to multiword expressions. We present a few representative examples in Table 5.

5.3. Lemmatization

As Wartena (2019) states, state-of-the-art lemmatizers for German are often rule-based, which we can attribute to the fact that lemmatization is an analytical step heavily depending on morphological analysis. To map a word form onto a lemma is to make a claim about how that word form is related to others on a linguistic level, and one must highlight the contrast between lemmatization and stemming here. For stemming, it is not necessary for the string to correspond to a word one can find in the lexicon, since it primarily aims at mapping related words to a common string to counteract data sparsity. However in lemmatization, we claim to map words to their lexicon form, which is why a non-existent lemma is always wrong whereas a stem that does not exist as a word can serve its purpose regardless. This is why linguistically sound lemmatization is foundational for tasks like identifying candidate words acting as metaphors, where we depend on true lexicon forms with semantic meaning.

This contrast made, we turn our attention to statistical lemmatization models and find that they are often not as reliable as their quantitative performance metrics suggest. While Stanza (Qi et al., 2018) reports F_1 = 97.23 % to 98.04 %[8] for their default

[7] Here, we follow Stanza's internal evaluation approach yielding character-level F1 scores.

[8] combined_charlm: https://stanfordnlp.github.io/stanza/performance.html#system-performance-on-ud-treebanks

Gold	Easy	Hard
Jer_7,18	Jer_7,18	Jer ; 7,18
Lukas_15,11-24	Lukas_15,11-24	Lukas ; 15,11-24
2._Buch_Mose	2._Buch_Mose	2. ; Buch ; Mose

Table 5: Examples of tokenization of Biblical citations in *easy* vs. *hard* scenarios as produced by our custom tokenization model. ";" represents the beginning of a new token.

Sermons		GSD		
#Sent	#Char	#Sent	#Char	F1-val
12,103	927,614	13,813	1,388,381	57.11
12,103	927,614	6,906	656,466	**99.99**

Table 6: Training mixture for Stanza tokenizer model with varying proportions of GSD support reporting number of sentences (#Sent) and characters (#Char) per dataset and F1 score on the validation set of sermons.

Form	Stanza	spaCy
Jesu	Jesu	Jesu
Jesus	Jesus	jesus
Gottes	Gott	gott
Bergpredigt	Bergpredigt	bergpredigen
Anfällen	Anfall	anfällen
reifen	reif	reifen
trete	treten	tren
5a	zeichnen	5a
Geist	Geist	isen
dass	dass	daß
dass	dass	dass
am	am	an
ihm	er	ihm
mir	ich	mir
aller	alle	aller
allem	alle	alle
was	was	wer

Table 7: Illustrating examples of unreliable performance of neural SOTA models on German lemmatization reporting token form, POS tag according to STTS, Stanza output, and spaCy output.

lemmatization model for German and spaCy (Honnibal et al., 2020) 98.00 % accuracy[9], Table 7 illustrates how high performing models fail at cases that are not inherently difficult or "just" domain-specific. We observe four classes of common mistakes:

1. Nouns specific to the religious domain such as *Jesus* or *Bergpredigt* 'Sermon on the Mount', which are either just lowercased or lemmatized as verbs by spaCy.
2. Non-sensical lemmas for various classes such as *5a* → *zeichnen* 'draw' or *Geist* 'spirit' → *isen*.
3. Unreliable predictions, where the same model produces different outputs for arguably the same input. For example, spaCy predicts *daß* for *dass* in some cases and *dass* correctly in others.
4. Classes where outputs differing between models can be attributed to different guidelines, e.g. how to lemmatize preposition–article contractions ("Schmelzformen") and pronouns.

5.3.1. Methods

We target each of the above types of errors with our implementation of GermaLemma++ (GL++), an ensemble of rule-based, lookup-based, and neural lemmatization methods and follow in the footsteps of Ortmann (2019), who adapts GermaLemma (Konrad, 2019) with additional rules for pronoun handling and introduces spaCy as a large-scale fallback to the original rule-based model. For GL++, we propose extensive handling of proper nouns, allow for domain adaptation, add additional rules for pronouns, adjectives, adverbs, articles, as well as conjunctions, and optimize the model choice for the fallback. The new GL++ architecture is illustrated in Figure 1 in the appendix.

Both preceding versions propose the original word form as the lemma for proper nouns, i.e. words that receive the POS tag *NE* from the Stuttgart-Tübingen-Tagset (STTS; Schiller et al., 1999). Cases such as *Jesu* or *Gottes* suggest that this solution is not always suitable. We utilize Kaikki (Ylonen, 2022), a digital archive providing 13 631 German proper nouns, and provide GL++ with a look-up table based on this resource. Only when the word is out-of-vocabulary do we assign the original word form as the lemma.

Since GermaLemma already has a dictionary-

[9] de_core_news_sm: https://spacy.io/models/de

based component based on the TIGER corpus, we exploit this method further for domain adaptation and provide a Python interface to add lists of custom mappings to the dictionary. Since we are working with sermons, we additionally provide a dictionary already adapted to the religious domain.

We extend the rule-based pronoun handling of Ortmann (2019) with additional rules for pronouns, corresponding to the STTS tags *PIAT*, *PIS*, *PDAT*, *PWS*, and *PRELS* according to the guidelines presented in Section 4.2. Furthermore, we adopt a rule-based and non-invasive approach to the preposition–article contractions as well as conjunctions to leave them unchanged and map conventionalized adjective constructions such as *desweiteren* and *imfolgenden* to *weiter* and *folgend* respectively.

Ortmann (2019) proposes spaCy as a rule-based fallback model for cases where neither the original GermaLemma nor the additional rules could assign a lemma. However qualitative observations as in Table 7 suggest that these edge cases might be exactly what spaCy often fails at. We conduct a quantitative analysis contrasting GL++ with spaCy and with Stanza as a fallback. Manually curating an evaluation set of 19 145 tokens shows that the latter provides the correct lemma in 34 and the former in only 11 out of 51 cases where the models differed in their predictions. Since the number of differing predictions is not high enough to make a definite judgement, we provide an option for choosing between the two fallback models in our GL++ implementation, but proceed with Stanza as the fallback for our corpus of sermons.

5.3.2. Results

For the final evaluation, we manually curate a set of 200 words from unseen sermons and contrast the performance in lemmatization of the default models provided by Stanza and spaCy vs. GL++. The 200 words are randomly selected content words whose lemmatization differs between models from 20 different sermons each providing 10 randomly chosen words.

We provide two gold standard versions: (i) “strict”: the lemma form that complies with our guidelines; (ii) “flexible”: other lemma forms that comply with the other guidelines. Table 9 shows that spaCy performs by far the worst in both conditions. Stanza clearly delivers the best results, even in the “strict” condition, for which GL++ is specifically designed. If we turn to accuracy separated by part of speech (see Table 10), we obtain that all models perform similarly well with adjectives. spaCy’s problems lie mainly with nouns and verbs. GL++ achieves exactly the same results as Stanza with adjectives and verbs, but performs significantly worse with nouns.

In order to assess the severity of the deviations in the lemmas, we classify errors into four types, see Table 8. For types 1–3, the corresponding lemmas can still be guessed (e.g., by a human annotator), whereas this is no longer possible for type 4. The critical errors are therefore the “strong” errors, as these typically involve the assignment of completely incorrect lemmas. Table 11 shows how often each error type occurs in the different models. The “strong” type occurs most frequently with spaCy, but also with the other models. Table 12 shows some examples of the different models. spaCy (and also GL++) quite often produce non-words, while Stanza more often assigns incorrect existing lemmas.

6. Conclusion

In this paper, we introduced a new corpus of contemporary German sermons and described some of the fundamental corpus preparation steps that we have taken, for which we adopt a semi-automatic annotation approach. We described the annotation guidelines that we applied in this corpus, which reflect the specific properties of this text type. Our analysis of established tools for tokenization, sentence segmentation, and lemmatization shows that these tasks are not yet as solved as they often seem and that it is worthwhile to approach them in a domain-sensitive way.

Our evaluation of domain-adaptable neural models for tokenization and sentence segmentation show that such models can be successfully adapted to our sermon-specific guidelines, but only to a certain extent. In both tasks, there are certain error classes that remain persistent. With respect to lemmatization, Stanza appears to provide good overall lemmatization performance, though the error rates are still quite high. A frequent error is the generation of non-existent lemmas: Lemmatizers should do more to avoid non-words, ideally linking to specific entries in a lexicon as opposed to providing strings alone, which are often ambiguous. If there is no clear solution, a model should not predict any lemma, and this would do less damage than making a misleading prediction.

In general, we wish to emphasize the importance of a domain-specific approach for more correct annotations, since different domains have different properties and requirements, and different corpora are structured in different ways. The pursuit of high-quality annotation is particularly important in these early stages of corpus preparation, since errors here will propagate through all subsequent stages of processing, which can then have unpredictable effects on derived automatic annotations as well as on any statistical analysis one carries out on the basis of such corpus data.

Error Type	Description	Example
marginal	Incorrect capitalization Or correct stem (vowel + consonants) with incorrect inflection but nom. sg.	*Striche* → **strich* /✓ *Strich* *rechte* → **rechter* /✓ *recht*
small	Correct stem, incorrect inflection	*Abschieden* → **Abschiede* /✓ *Abschied*
medium	Incorrect stem but lemma can be guessed	*blies* → **blies* /✓ *blasen*
strong	Lemma distorted, e.g., other, unrelated lemma	*Zelt* → **zeln* /✓ *Zelt*

Table 8: Error types of incorrect lemma forms.

Model	Strict	Flex
spaCy	0.26	0.38
Stanza	0.72	0.81
GL++	0.64	0.69

Table 9: Accuracies (strict & flexible) per model.

Model	NOUN	ADJ	VERB
spaCy	0.29	0.76	0.23
Stanza	0.86	0.73	0.82
GL++	0.65	0.73	0.82

Table 10: Accuracies (flexible) per model and main parts of speech.

Word form	✓Lemma	* Lemma
spaCy:		
abzugleiten	abgleiten	Abzugleit
mitgeprägt	mitprägen	mitgepragen
begehrenswert	begehrenswert	begehrenswern
Zelt	Zelt	zeln
Stanza:		
http://wiki.vol[...]	(same)	Gewinntomaniet
betrüge	betrügen	betragen
Wollen	Wollen	Wolle
GL++:		
bitter	bitter	bitt
abzugleiten	abgleiten	abzugleien
Wüsst	wissen	wüsen

Table 12: Example of strong deviations of lemmas.

7. Acknowledgements

This research is funded by the Deutsche Forschungsgemeinschaft (DFG) SFB1475 – project number 441126958. Cora Haiber is supported by the Konrad Zuse School of Excellence in Learning and Intelligent Systems (ELIZA) through the DAAD programme Konrad Zuse Schools of Excellence in Artificial Intelligence, sponsored by the Federal Ministry of Education and Research.

Furthermore, we thank Tamara Abdul Majeed and Sophie Kratz for their efforts in manual annotation and correction.

Model	marg.	small	medium	strong
spaCy	10	20	78	17
Stanza	2	10	12	10
GL++	3	37	15	7

Table 11: Frequencies of error types per model.

8. Bibliographical Referencess

Michael Beißwenger, Sabine Bartsch, Stefan Evert, and Kay-Michael Würzner. 2015. Richtlinie für die manuelle Tokenisierung von Sprachdaten aus Genres internetbasierter Kommunikation: Guideline document from the Empirikom shared task on automatic linguistic annotation of internet-based communication (EmpiriST 2015).

Toms Bergmanis and Sharon Goldwater. 2018. Context sensitive neural lemmatization with Lematus. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1391–1400, New Orleans, Louisiana. Association for Computational Linguistics.

Emanuel Borges Völker, Maximilian Wendt, Felix Hennig, and Arne Köhn. 2019. HDT-UD: A very large Universal Dependencies treebank for German. In *Proceedings of the Third Workshop on Universal Dependencies (UDW, SyntaxFest 2019)*, pages 46–57, Paris, France. Association for Computational Linguistics.

Sabine Brants, Stefanie Dipper, Peter Eisenberg, Silvia Hansen, Esther König, Wolfgang Lez-

ius, Christian Rohrer, George Smith, and Hans Uszkoreit. 2004. TIGER: Linguistic interpretation of a German corpus. *Journal of Language and Computation*, 2(4):597–620.

Berthold Crysmann, Silvia Hansen-Schirra, George Smith, and Dorothea Ziegler-Eisele. 2005. *TIGER Morphologie-Annotationsschema*. Projekt TIGER, Universität des Saarlandes, Universität Stuttgart, Universität Potsdam.

Tom De Smedt and Walter Daelemans. 2012. Pattern for Python. *Journal of Machine Learning Research*, 13(66):2031–2035.

Nils Diewald, Marc Kupietz, and Harald Lüngen. 2022. Tokenizing on scale: Preprocessing large text corpora on the lexical and sentence level. In *Dictionaries and Society. Proceedings of the XX EURALEX International Congress*, pages 208–221, Mannheim. IDS-Verlag.

Stefanie Dipper and Ronja Laarmann-Quante. 2024. UD for German poetry. In *Proceedings of the 4th International Conference on Natural Language Processing for Digital Humanities*, pages 177–188, Miami, USA. Association for Computational Linguistics.

Aleksei Dorkin and Kairit Sirts. 2024. Comparison of current approaches to lemmatization: A case study in Estonian. _eprint: 2404.15003.

Markus Frohmann, Igor Sterner, Ivan Vulić, Benjamin Minixhofer, and Markus Schedl. 2024. Segment Any Text: A universal approach for robust, efficient and adaptable sentence segmentation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11908–11941, Miami, Florida, USA. Association for Computational Linguistics.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength natural language processing in Python.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. LoRA: Low-rank adaptation of large language models. *ArXiv*, abs/2106.09685.

Tibor Kiss and Jan Strunk. 2006. Unsupervised multilingual sentence boundary detection. *Computational Linguistics*, 32(4):485–525.

Markus Konrad. 2019. GermaLemma: A lemmatizer for German language text. Publication Title: GitHub repository.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 27th International Conference on Neural Information Processing Systems – Volume 2*, NIPS'13, pages 3111–3119. Curran Associates Inc.

George A. Miller. 1994. WordNet: A lexical database for English. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8–11, 1994*.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666, Portorož, Slovenia. European Language Resources Association (ELRA).

Katrin Ortmann. 2019. GermaLemma++. Publication Title: GitHub repository.

Katrin Ortmann, Adam Roussel, and Stefanie Dipper. 2019. Evaluating off-the-shelf NLP tools for German. In *Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019): Long Papers*, pages 212–222, Erlangen, Germany. German Society for Computational Linguistics & Language Technology.

Thomas Proisl and Peter Uhrig. 2016. SoMaJo: State-of-the-art tokenization for German web and social media texts. In *Proceedings of the 10th Web as Corpus Workshop (WAC-X) and the EmpiriST Shared Task*, pages 57–62, Berlin. Association for Computational Linguistics.

Peng Qi, Timothy Dozat, Yuhao Zhang, and Christopher D. Manning. 2018. Universal Dependency parsing from scratch. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 160–170, Brussels, Belgium. Association for Computational Linguistics.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. _eprint: 2003.07082.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Carlos Ramisch. 2015. *Multiword Expressions Acquisition: A Generic and Open Framework*. Springer International Publishing.

Anne Schiller, Simone Teufel, Christine Stöckert, and Christine Thielen. 1999. Guidelines für das Tagging deutscher Textcorpora mit STTS (Kleines und großes Tagset). *Technischer Bericht. Universität Stuttgart, Institut für Maschinelle Sprachverarbeitung*.

Heike Telljohann, Erhard W. Hinrichs, Sandra Kübler, Heike Zinsmeister, and Kathrin Beck. 2017. *Stylebook for the Tübingen Treebank of Written German (TüBa-D/Z)*. Seminar für Sprachwissenschaft, Universität Tübingen.

Olia Toporkov and Rodrigo Agerri. 2024. On the role of morphological information for contextual lemmatization. *Computational Linguistics*, 50(1):157–191.

Christian Wartena. 2019. A probabilistic morphology model for German lemmatization. In *Proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019)*, pages 40–49. Fakultät III – Medien, Information und Design.

Tatu Ylonen. 2022. Wiktextract: Wiktionary as machine-readable structured data. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1317–1325, Marseille, France. European Language Resources Association.

A. Appendix

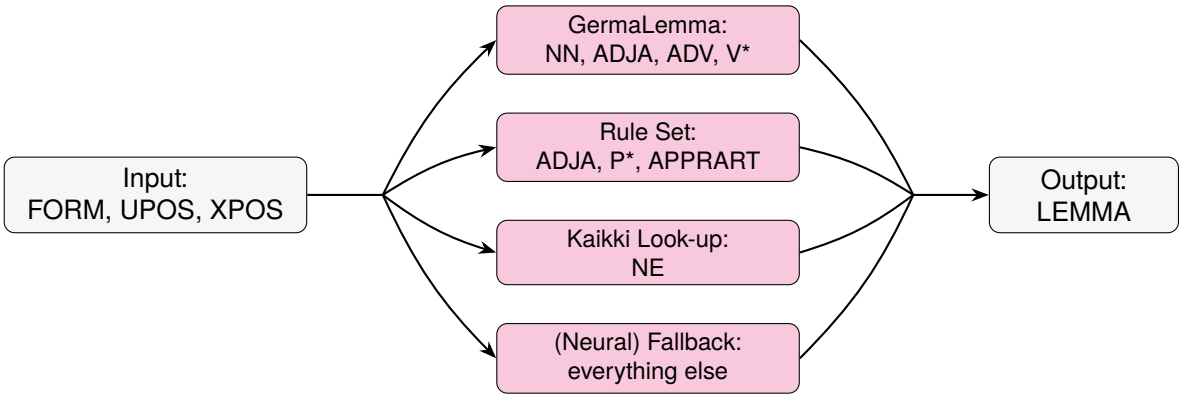


Figure 1: GL++ ensemble combining rules, lookup, and a neural model.

Semantic, Syntactic, Lexical: What Makes QA Augmentation Work in Limited Quantity?

Benedictus Kent Rachmat[1,2], Thomas Gerald[1], Takuya Nakamura[1], Zheng Zhang[2], Cyril Grouin[1]
[1]Université Paris-Saclay, CNRS, LISN, Orsay, France
[2]Embedded AI Lab, SLB, Clamart, France
{rachmat, gerald, nakamura, grouin}@lisn.fr, zhang@slb.com

Abstract

Data augmentation is a common fix in domains where training data is scarce or difficult to collect, such as specialized medical or any other domain specific applications. In question answering (QA), most studies report headline accuracy while saying little about the quality of the synthetic data. Here, quality goes beyond fluent rewording: augmented items must remain faithful to the supporting evidence and preserve the original answerability. We study three augmentation families *lexical*, *syntactic*, and *semantic* edits generated with LLaMA 3.1 70B, and analyze how these edits affect model behavior. To mirror low-resource settings, we focus on subsets of SQuADv2 (general) and PubMedQA (biomedical, domain specific). We report Exact Match (EM)/F1 alongside quality diagnostics, yielding a fuller picture than accuracy alone. Our results show that augmentation behaves differently across domains and scales. In SQuADv2, augmented variants maintain performance on par with baselines, showing that added diversity mostly does not harm model quality, whereas in PubMedQA semantic edits bring improvements under extreme scarcity and support stronger performance as supervision grows.

Keywords: Question Answering, Evaluation Metrics, Human Validation, Diversity, Synthetic Data, Domain Specific, Data Augmentation

1. Introduction

Data augmentation offers a straightforward promise to generate additional training examples to compensate for scarce annotations and improve model generalization. This promise has been realized in many NLP tasks, especially classification and text generation (Feng et al., 2021; Dai et al., 2025), where methods like paraphrasing and back-translation yield consistent gains (Sobrevilla Cabezudo et al., 2024). By exposing models to varied phrasing and diverse examples, augmentation can improve robustness, particularly in low-resource settings where labeled data is scarce. Large language models (LLMs) further enable synthetic corpus creation at scale, but raise a key question: what kind of linguistic variation do they actually produce, and which kind is most useful for downstream learning?

In question answering (QA), augmented examples must satisfy constraints beyond surface fluency, and the edits an augmentation system can apply to a question–answer pair fall into three linguistically motivated families. *Lexical* augmentation replaces individual words with synonyms or near-synonyms (e.g., *meet* → *encounter*), varying surface vocabulary while preserving meaning and structure. *Syntactic* augmentation restructures the sentence. For instance, converting active to passive voice, reordering clauses or changing grammatical form but not propositional content. *Semantic* augmentation goes further: given the same source passage, it generates an entirely new question targeting a different fact. These three levels mirror classical linguistic analysis (lexicon, syntax, semantics) and allow us to isolate which kind of variation is most beneficial for model training.

In contextual QA, the system must extract an answer span from a given passage or return "unanswerable" if none exists, so augmented examples must remain faithful to the passage and preserve answerability. Prior work has emphasized scale, as large synthetic sets often boost exact match and F1 (Alberti et al., 2019; Shakeri et al., 2020), yet without quality control, such gains may be misleading artifacts. We study this on two contrasting benchmarks: SQuADv2 (Rajpurkar et al., 2018), a general-domain dataset built from Wikipedia with both answerable and unanswerable questions, and PubMedQA (Jin et al., 2019), a biomedical dataset derived from PubMed abstracts requiring a categorical label (*yes*/*no*/*maybe*) and a free-text justification. Rather than using full training sets, we sample small subsets to mimic low-resource conditions, as truly low-resource QA datasets tend to be noisy and under-explored (Castelli et al., 2020; Jin et al., 2022); these well-established benchmarks provide a controlled testbed for studying how augmentation generalizes across domains and data scales.

Contributions. Our main contributions are:

- Cross-domain study: to our knowledge, the first systematic comparison of lexical, syntactic, and semantic augmentation across SQuADv2 (general) and PubMedQA (biomedical)

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 224–236

- Augmentation pipeline: we design a controlled setup where each augmentation type is generated and validated with GPT-4o, with a human audit confirming strengths and failure modes
- Scaling analysis: we reveal non-monotonic behavior across supervision scales, linking overfitting at mid-size subsets to LoRA dynamics
- Error taxonomy: we provide a structured taxonomy of augmentation failures, offering diagnostic tools for improving future QA augmentation

2. Related Work

Data augmentation for QA operates at different linguistic levels. Lexical methods such as synonym substitution expand vocabulary diversity (Wei and Zou, 2019). Syntactic approaches rephrase questions via structural changes like back-translation (Sennrich et al., 2016), improving robustness to phrasing (Yu et al., 2018). Semantic augmentation leverages generative models to create new QA pairs or richer paraphrases, yielding strong gains in benchmarks like SQuADv2 and PubMedQA (Alberti et al., 2019; Shakeri et al., 2020; Guo et al., 2023). While foundational studies showed improvements, later work highlighted mixed results: for example, back-translation sometimes fails to transfer across domains (Longpre et al., 2019).

LLM-based augmentation. Recent work has increasingly leveraged LLMs as data generators. Schmidt et al. (2024) show that LLM prompting generates high-quality synthetic QA data that improves few-shot performance. Chowdhury and Chadha (2024) demonstrate that LLM-generated augmentation improves distributional robustness across multiple QA datasets. Chan et al. (2024) categorize synthetic data strategies into answer augmentation, question rephrase, and new question generation, a taxonomy that parallels our lexical/syntactic/semantic distinction and show the optimal strategy depends on the seed-to-query ratio. In domain-specific settings, Khlaut et al. (2024) use GPT-4 to generate medical QA training data from textbooks, achieving competitive performance with smaller fine-tuned models.

Quality control and synthetic data risks. Scaling augmentation without fidelity may harm rather than help. Chen et al. (2024) identify that uniform format in synthetic QA pairs leads to pattern overfitting and distribution shifts. Pletenev et al. (2025) study knowledge packing in LoRA adapters on Llama-3.1-8B-Instruct, finding that biased synthetic training data causes regression to overrepresented answers. For quality validation, the LLM-as-a-Judge paradigm (Zheng et al., 2023; Gu et al., 2025) has shown that GPT-4 achieves $>$80% agreement with human experts, though position and verbosity biases persist.

Positioning. Our work integrates lexical, syntactic, and semantic edits in one framework with a validation loop, combining the systematic comparison of augmentation types with quality-aware filtering. Unlike prior work that isolates single strategies, we study their interactions across data scales and domains.

3. Methodology

3.1. Datasets

We evaluate on two extractive QA benchmarks: SQuADv2 (Rajpurkar et al., 2018), also known as SQuADRUn, a general-domain reading comprehension dataset that extends the original SQuAD (Rajpurkar et al., 2016) with unanswerable questions, and PubMedQA (Jin et al., 2019), a biomedical QA dataset derived from PubMed abstracts. This pairing lets us assess whether augmentation strategies transfer across domains and linguistic distributions. To simulate low-resource settings, we downsample each training set to $\frac{1}{32}$ of its original size as the base augmentation pool, then sample progressively smaller nested fractions ($\frac{1}{128} \subset \frac{1}{64} \subset \frac{1}{32} \subset \frac{1}{16}$). Test sets remain fixed (11,873 for SQuADv2, 1,000 for PubMedQA). Since neither benchmark provides an official dev set, we reserve 10% of the original training data as a validation pool, and consistently hold out 10% of each training portion for fine-tuning validation. For SQuADv2, we apply stratified sampling to preserve the answerable/unanswerable ratio. Each downsampled split therefore contains both answerable and unanswerable items; the unanswerable questions are carried into augmentation.

Fraction	SQuADv2	PubMedQA
$1/128$	1,015	1,648
$1/64$	2,031	3,296
$1/32$	4,062	6,593
$1/16$	8,125	13,187

Table 1: Training sample counts per supervision scale

To characterize the data geometry, we project samples into embedding space using Qwen3-Embedding-0.6B. As shown in Figure 1, the two corpora exhibit clear domain separation, though some SQuADv2 passages with biomedical content cluster near PubMedQA and vice versa, indicating the embedding model captures cross-domain se-

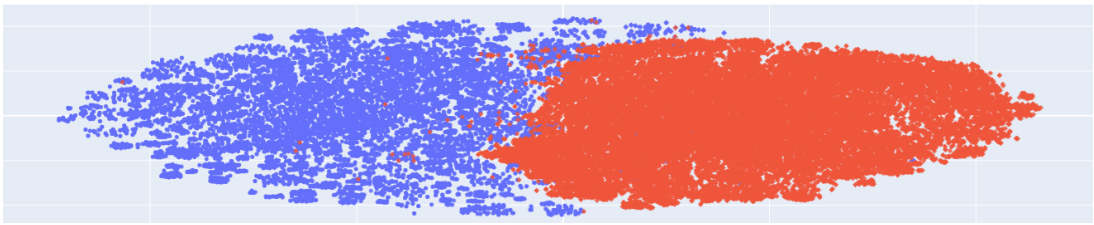


Figure 1: Embedding visualization showing separation between general-domain (blue, SQuADv2) and biomedical (red, PubMedQA) samples, with overlap pockets reflecting cross-domain similarity

mantic proximity while preserving broader category distinctions.

3.2. Data Augmentation

Prior work has shown that large models can be powerful engines for creating diverse training corpora (Puri et al., 2020; Wang et al., 2023; Mitra et al., 2024). While many black-box augmentation pipelines have proven effective in practice, our goal is more focused within a linguistic scope, aiming to maximize the contextual knowledge and generative ability of LLMs to produce questions or question–answer pairs from multiple linguistic angles. To isolate this effect, we employ a single model, `meta-llama/Llama-3.1-70B-Instruct`,[1] and keep the prompting setup consistent for both generation and inference. This design ensures that observed differences stem from the type of linguistic augmentation rather than engineering optimizations. In this way, we do not attempt to beat state-of-the-art accuracy, but rather to analyze and better understand the role of linguistic augmentation itself. The exact prompts used for each augmentation type are provided in Appendix A. SQuADv2 contains unanswerable items, which we retain in the augmentation pool. Lexical and syntactic edits preserve the unanswerable status, while semantic augmentation generates new questions grounded in the passage, marking them as unanswerable when no valid span exists. This concern does not apply to PubMedQA, which has no unanswerable category.

Within the $1/32$ supervision split, each original instance is expanded by calling the model twice per augmentation type (lexical, syntactic, semantic), producing two variants per category. Thus, one original question yields six synthetic counterparts. We then apply GPT-4o to validate whether the generated examples respect the intended category constraints.[2] After validation, we retain only the valid examples for fine-tuning. Table 2 summarizes the distribution of validation outcomes across datasets.

Label	SQuADv2	PubMedQA
Valid	17,768	19,050
Unsure	4	17
Invalid	7,207	5,927

Table 2: Validation outcomes for synthetic QA pairs across datasets. Only the *valid* examples are used for downstream fine-tuning

3.2.1. Lexical-based data augmentation

The question is rephrased by replacing words with synonyms, near-synonyms, or idiomatic expressions. The answer remains unchanged, but the surface form of the question differs.

- Original question: *At what age did Beyoncé meet LaTavia Robertson?* (SQuADv2)
- Lexical variant: *At what age did Beyoncé encounter LaTavia Robertson?*

3.2.2. Syntactic-based data augmentation

This strategy rewrites questions using alternate grammatical structures such as switching between active and passive voice or simple and complex forms while preserving both meaning and the original answer.

- Original question: same as above
- Syntactic variant: *LaTavia Robertson first met Beyoncé when she was how old?*

3.2.3. Semantic-based data augmentation

Unlike the above, semantic augmentation produces new question-answer that probe different aspects of the same passage. These are not strict paraphrases: answers may differ, yet they remain grounded in the same evidence.

[1] https://huggingface.co/meta-llama/Llama-3.1-70B-Instruct

[2] We intentionally oversample at this stage, since a portion of synthetic examples are filtered out; we need to ensures that the final supervision proportions remain consistent across categories.

- Original context: *At age eight, Beyoncé and childhood friend Kelly Rowland met LaTavia Roberson [...] They were placed into a group with three other girls as Girl's Tyme, and rapped and danced on the talent show circuit in Houston. After seeing the group, R&B producer Arne Frager brought them to his Northern California studio and placed them in Star Search, the largest talent show on national TV at the time. Girl's Tyme failed to win [...]*
- Semantic variant: *What was the name of the talent show that Girl's Tyme failed to win?*
- Answer: *Star Search*

3.3. Models and Fine Tuning

We fine-tune two open-source instruction-following LLMs: `Llama-3.1-8B-Instruct` (Llama Community License) and `Qwen2.5-7B-Instruct` (Apache 2.0), abbreviated hereafter as Llama8B-I and Qwen7B-I respectively (with Llama70B-I denoting Meta-Llama-3.1-70B-Instruct used for augmentation). Choosing open-source backbones ensures reproducibility and alignment with community-driven policies. Fine-tuning uses LoRA (Hu et al., 2022) with rank r=16, α=16, and dropout 0.05, applied to all attention and feed-forward projection matrices, updating only 0.53% of parameters. We use AdamW with a learning rate of 1×10^{-4}, cosine schedule, warmup ratio 0.1, weight decay 0.01, and a per-device batch size of 32 with 4 gradient accumulation steps. Training runs for 4 epochs in bf16 precision. Full hyperparameters are listed in Appendix B. To set the training schedule, we experimented on the $\frac{1}{128}$ split with three seeds: evaluation loss diverged after epoch 4 and extending to 8 epochs yielded no gains, so we fix training to 4 epochs. Early trials also showed that generating multiple QA pairs per call caused hallucinations (Nayab et al., 2025); switching to one pair per call improved quality.

3.4. Experimental Workflow

We construct augmented training sets by mixing the original data with different types of linguistic edits *lexical*, *syntactic*, *semantic*, and a combined *all* variant while keeping the overall sample size fixed across conditions. After generating augmented data, we perform a quality-control step: each candidate QA pair is validated by `GPT-4o` (Hurst et al., 2024) acting as an LLM judge (Gu et al., 2025), which assesses whether the augmented items remain faithful to their supporting context and relevant to their edit types. In Section 5, we further compare the alignment of `GPT-4o` judgments with human annotations. From this filtered subset, we construct the final training data used for fine-tuning. In total, our setup covers:

- 2 datasets: SQuADv2 (general) and PubMedQA (biomedical)
- 2 models: Llama8B-I and Qwen7B-I
- 3 scales: downsampled subsets at $\frac{1}{128}$, $\frac{1}{64}$, and $\frac{1}{32}$ of the original training set (we omit $\frac{1}{16}$ due to time constraints, but fine-tune the baseline at this scale to enable comparison with lower-proportion counterparts)
- 5 augmentation conditions: baseline (no augmentation, purely train set), syntactic, lexical, semantic, and all (uniform mixture of the other categories)

This factorial design yields $2 \times 2 \times 3 \times 5 = 60$ runs overall. Together, these experiments allow us to systematically analyze how augmentation type, data scale, and model choice interact under low-resource conditions.

4. Evaluation

We assess model outputs with three metrics: *Exact Match (EM)*, *word-level F1*, and *semantic similarity* computed from sentence embeddings. Together, these capture strict string agreement, partial overlap, and meaning-level alignment. For PubMedQA, we substitute a simpler EM (*EM bin.*), which evaluates whether the predicted label (`yes`, `no`, or `maybe`) matches the gold annotation. At the same time, we retain word-level F1 and semantic similarity to evaluate the longer free-form reasoning answers, since they capture partial overlap and meaning beyond the discrete label. We do not compute span-level EM on PubMedQA, as answers are long sentences that rarely match exactly even GPT-4o achieves zero EM in this setting.

Normalization and multiple references. Following standard practice for contextual QA, predictions and references are normalized by lower-casing, removing punctuation and articles, and collapsing extra whitespace.

SQuADv2 Analysis. Our evaluation on SQuADv2, summarized in Table 3, reveals several key insights. First, supervised fine-tuning provides a performance boost over the vanilla models (i.e., base models used without any task-specific fine-tuning). For instance, the Qwen7B-I baseline, when fine-tuned on just the $1/128$ data subset, achieves 0.68 EM and 0.77 F1, outperforming its vanilla counterpart's 0.54 EM and 0.65 F1. This highlights the critical value of task-specific adaptation, even with minimal data. When comparing models, the fine-tuned

Model	EM	F1	Sim ↑
SQuADv2 (vanilla, i.e., no fine-tuning)			
Qwen7B-I	0.54	0.65	0.73 ± 0.38
Llama8B-I	0.57	0.67	0.74 ± 0.39
Llama70B-I	0.54	0.66	0.74 ± 0.38
GPT-4o (vanilla)	0.56	0.68	0.78 ± 0.33
SQuADv2 (1/128 subset)			
Qwen7B-I Baseline	0.68	0.77	0.83 ± 0.33
Qwen7B-I Semantic	0.69	0.77	0.83 ± 0.33
Qwen7B-I Syntactic	0.68	0.77	0.83 ± 0.33
Qwen7B-I Lexical	0.68	0.77	0.83 ± 0.33
Qwen7B-I All	0.68	0.77	0.83 ± 0.33
Llama8B-I Baseline	0.63	0.72	0.78 ± 0.37
Llama8B-I Semantic	0.60	0.70	0.76 ± 0.39
Llama8B-I Syntactic	0.62	0.71	0.77 ± 0.38
Llama8B-I Lexical	0.62	0.72	0.78 ± 0.38
Llama8B-I All	0.62	0.71	0.77 ± 0.38
SQuADv2 (1/64 subset)			
Qwen7B-I Baseline	0.72	0.79	0.84 ± 0.33
Qwen7B-I Semantic	0.68	0.76	0.82 ± 0.35
Qwen7B-I Syntactic	0.72	0.79	0.84 ± 0.33
Qwen7B-I Lexical	0.71	0.79	0.84 ± 0.33
Qwen7B-I All	0.71	0.79	0.84 ± 0.33
Llama8B-I Baseline	0.67	0.73	0.77 ± 0.40
Llama8B-I Semantic	0.60	0.69	0.75 ± 0.40
Llama8B-I Syntactic	0.65	0.73	0.77 ± 0.39
Llama8B-I Lexical	0.66	0.73	0.77 ± 0.40
Llama8B-I All	0.65	0.73	0.77 ± 0.39
SQuADv2 (1/32 subset)			
Qwen7B-I Baseline	0.57	0.67	0.74 ± 0.39
Qwen7B-I Semantic	0.55	0.66	0.76 ± 0.37
Qwen7B-I Syntactic	0.50	0.62	0.70 ± 0.40
Qwen7B-I Lexical	0.49	0.59	0.68 ± 0.42
Qwen7B-I All	0.52	0.63	0.71 ± 0.40
Llama8B-I Baseline	0.58	0.61	0.63 ± 0.47
Llama8B-I Semantic	0.55	0.64	0.70 ± 0.43
Llama8B-I Syntactic	0.57	0.60	0.64 ± 0.46
Llama8B-I Lexical	0.56	0.60	0.63 ± 0.46
Llama8B-I All	0.60	0.65	0.69 ± 0.44
SQuADv2 (1/16 subset)			
Qwen7B-I Baseline	0.73	0.80	0.85 ± 0.33
Llama8B-I Baseline	0.70	0.77	0.80 ± 0.36

Table 3: Evaluation results on SQuAD v2 subsets with different augmentation strategies

Qwen7B-I consistently outperforms Llama8B-I across most baseline settings, suggesting stronger data efficiency for Qwen on this benchmark. Surprisingly, our data augmentation strategies (Lexical, Syntactic, Semantic, All) offer no substantial benefit. Most of the variants perform identically to the baseline. This indicates that augmentation does not substantially alter performance. Notably, augmented models match the baseline despite seeing fewer unique source passages, suggesting that increased QA diversity compensates for reduced contextual breadth.

The relationship between data proportion and performance is notably non-monotonic. While results improve when scaling from the $1/128$ to the $1/64$ subset, they drop unexpectedly at $1/32$ before recovering at $1/16$. We hypothesize that this dip may reflect an interaction between dataset size and the dynamics of LoRA fine-tuning. With very small subsets (e.g., $1/128$), the model remains underfit but relatively stable; with larger subsets (e.g., $1/16$), the adapter has enough signal to generalize effectively. At intermediate scales, however, the model may cross a threshold where it overfits to limited but noisy supervision, leading to degraded performance. Such non-monotonic scaling behavior has been observed in other settings (Nakkiran et al., 2021), suggesting that careful calibration of data size and fine-tuning strategy is essential in low-resource environment.

Summary. For Qwen7B-I, augmentation yields parity with the baseline across splits while using fewer unique source passages suggesting that added QA diversity can substitute for contextual breadth (no single strategy dominates).

PubMedQA Analysis. Our results on PubMedQA (Table 4) differ markedly from SQuADv2. A key distinction is that PubMedQA's evaluation includes EM (bin.) that only checks whether the model predicted the correct label (*yes*, *no*, or *maybe*) and the rest reflect how well the model reasons in free-text explanations. This separation reveals an important phenomenon, some models are "stubborn" producing an answer reasoning without committing to the correct discrete label, which lowers EM but leaves F1 and similarity scores intact. For instance, Qwen7B-I at the $1/128$ subset achieves only 0.75 EM but 0.53 similarity, while its semantic-augmented variant keeps EM stable but lifts similarity to 0.75. This suggests that label prediction and reasoning fluency are not always aligned, and both need to be considered together.

When comparing augmentation styles and data proportions, several signals emerge. Semantic augmentation proves most helpful at the smallest scale, for Qwen7B-I, it improves F1 from 0.18 to 0.28 at $1/128$ and raises similarity by more than 0.20. In contrast, syntactic and lexical variants are less reliable, occasionally dropping EM sharply (e.g., Qwen7B-I Syntactic at $1/32$ falls to 0.42 EM) despite stable reasoning metrics. The "All" strategy produces middling results, rarely surpassing semantic alone. Scaling with more supervision does not yield monotonic improvements, performance rises from $1/128$ to $1/64$, then becomes erratic at $1/32$, and collapses at $1/16$. This instability suggests an interaction between noisy biomedical su-

Model	EM (bin.)	F1	Sim ↑
PubMedQA (vanilla)			
Qwen7B-I	0.84	0.27	0.75 ± 0.09
Llama8B-I	0.88	0.26	0.73 ± 0.14
Llama70B-I	0.85	0.26	0.7 ± 0.22
GPT-4o (vanilla)	0.76	0.26	0.70 ± 0.23
PubMedQA (1/128 subset)			
Qwen7B-I Baseline	0.75	0.18	0.53 ± 0.35
Qwen7B-I Semantic	0.76	0.28	0.75 ± 0.14
Qwen7B-I Syntactic	0.76	0.19	0.56 ± 0.33
Qwen7B-I Lexical	0.72	0.19	0.54 ± 0.34
Qwen7B-I All	0.73	0.19	0.54 ± 0.34
Llama8B-I Baseline	0.89	0.18	0.49 ± 0.37
Llama8B-I Semantic	0.89	0.19	0.54 ± 0.34
Llama8B-I Syntactic	0.88	0.20	0.53 ± 0.36
Llama8B-I Lexical	0.88	0.19	0.53 ± 0.36
Llama8B-I All	0.89	0.14	0.38 ± 0.38
PubMedQA (1/64 subset)			
Qwen7B-I Baseline	0.71	0.31	0.75 ± 0.19
Qwen7B-I Semantic	0.86	0.24	0.68 ± 0.24
Qwen7B-I Syntactic	0.73	0.31	0.75 ± 0.20
Qwen7B-I Lexical	0.70	0.28	0.71 ± 0.26
Qwen7B-I All	0.82	0.24	0.64 ± 0.30
Llama8B-I Baseline	0.72	0.33	0.80 ± 0.09
Llama8B-I Semantic	0.88	0.18	0.52 ± 0.35
Llama8B-I Syntactic	0.71	0.33	0.80 ± 0.10
Llama8B-I Lexical	0.66	0.33	0.80 ± 0.09
Llama8B-I All	0.86	0.30	0.77 ± 0.13
PubMedQA (1/32 subset)			
Qwen7B-I Baseline	0.87	0.33	0.80 ± 0.09
Qwen7B-I Semantic	0.89	0.25	0.73 ± 0.12
Qwen7B-I Syntactic	0.42	0.33	0.81 ± 0.09
Qwen7B-I Lexical	0.56	0.33	0.79 ± 0.12
Qwen7B-I All	0.70	0.31	0.77 ± 0.16
Llama8B-I Baseline	0.38	0.33	0.80 ± 0.09
Llama8B-I Semantic	0.88	0.15	0.54 ± 0.26
Llama8B-I Syntactic	0.39	0.33	0.80 ± 0.09
Llama8B-I Lexical	0.39	0.33	0.80 ± 0.09
Llama8B-I All	0.82	0.33	0.80 ± 0.10
PubMedQA (1/16 subset)			
Qwen7B-I Baseline	0.36	0.33	0.8 ± 0.09
Llama8B-I Baseline	0.38	0.33	0.8 ± 0.09

Table 4: Evaluation results on PubMedQA subsets with different augmentation strategies

pervision and LoRA adaptation.

Model comparison reinforces these findings: Llama8B-I achieves strong EM at $1/128$ (0.89) but drops to 0.38 at $1/32$, while Qwen7B-I adapts better with more supervision, likely reflecting differences in tokenizer coverage and pretraining bias. Augmentation thus interacts with both model type and supervision size.

Summary. For PubMedQA, semantic augmentation is the only strategy that shows consistent gains, improving Qwen7B-I at the $1/128$ scale (higher F1 and similarity) without hurting EM. Syntactic and lexical variants are unstable, especially at $1/32$, while the mixed "All" setting rarely outperforms semantic alone. Llama8B-I performs best at extreme low-data ($1/128$), but Qwen7B-I adapts better at moderate scale ($1/32$).

5. Human Validation

Large language models can generate fluent but unfaithful content (Huang et al., 2025; Kalai et al., 2025), so we manually validate a subset of the synthetic QA data to quantify quality and calibrate an automatic screener. Our annotation pool comprises one Linguist, one PhD student in NLP, and two NLP Experts. For each dataset, we sample 75 items and assign one of three labels: *invalid*, *unsure*, and *valid*. The labeling guidelines differ slightly across augmentation types to reflect their specific failure modes (e.g., lexical vs. semantic drift). In parallel, we ask GPT-4o to label the same items using the exact guideline, in order to test whether it can reliably scale the validation to the full corpus.

5.1. Validation Protocol and Confidence Estimation

Protocol and timing. Annotators worked independently with the same instruction sheet and examples. After annotation, we measure the level of consensus between human raters and the GPT-4o outputs to assess whether the model can approximate human judgment. Manual validation is costly: 36–67 minutes per 75 SQuADv2 items and 1.5–3 hours for PubMedQA (domain-specific terms require slower reading). In contrast, GPT-4o labels the same batch in about 250 s (SQuADv2) and 370 s (PubMed).

Confidence Estimation from Log-Probabilities. To avoid repeated API calls for calibration, we approximate model confidence directly from the log-probability of the emitted label token ("-1", "0", or "1" referring to invalid, unsure, and valid labels). Specifically, we extract the log-prob reported for that token and convert it into a probability by exponentiation. This yields a single confidence score per item without additional queries. While approximate, this proxy reflects the model's internal preference for the chosen label and is commonly used in lightweight uncertainty estimation for LMs (Kadavath et al., 2022).

Figure 2 plots the distribution of GPT-4o confidence grouped by the human reference label. The model is highly confident overall, with mass near 1.0 for both *valid* and *invalid* cases. The valid class shows slightly broader spread (some outliers < 0.9), whereas invalid items are predicted with near-perfect certainty. Interestingly, the model never outputs the intermediate *unsure* label (as in our large-scale experiments, which yielded only a

negligible number of such cases) and thus shows no explicit uncertainty. This pattern indicates *overconfidence*: the model differentiates valid from invalid on average, but leaves little room for calibrated uncertainty, which cautions against fully automatic acceptance without human.

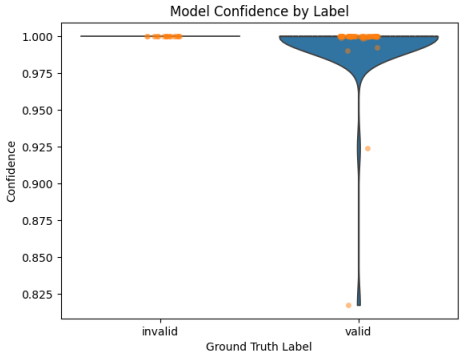


Figure 2: Model confidence (from label log-probs). Both datasets show the same pattern; we plot one dataset (SQuADv2) for brevity

5.2. Error Taxonomy from Human Review

Beyond a single validity label, annotators consistently flagged recurrent issues in synthetic QA. We group them into five categories:

- Grammar & fluency (bad grammatical phrasing)
- Faithfulness to context (missing key details or unsupported additions)
- Answer–question alignment (answer is over/under-specified relative to the question)
- Redundancy/minimal variation (near-duplicates or trivial rewrites)
- Clarity & specificity (vague wording, ambiguous acronyms, underspecified entities)

This taxonomy offers a structured lens for diagnosing weak points in synthetic QA. Going forward, these categories could be used not only for evaluation but also to guide augmentation itself. For instance, by adapting prompts to reduce redundancy, enforce grounding for faithfulness, or encourage clearer wording. They could also serve as automatic filter signals, training lightweight detectors that flag likely errors before human review.

5.3. Human vs. Model Validation

An important question is how closely model-based validation aligns with human experts, and whether this alignment shifts across domains. Manual review is slow and costly, whereas models like GPT-4o are efficient but may overlook domain subtleties. We therefore compare validity rates, annotator confidence, and category-level differences between humans and the model.

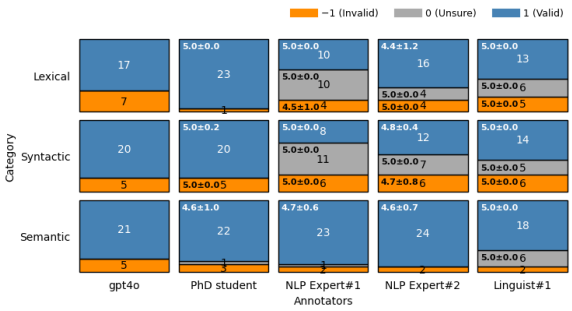


Figure 3: SQuADv2 validation (75 QA pairs). Agreement is relatively high, with disagreements concentrated in syntactic vs. semantic categories

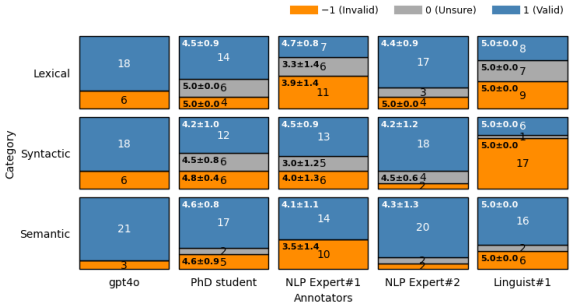


Figure 4: PubMedQA validation (75 QA pairs). NLP Experts marked more items as *invalid* or *unsure*, highlighting domain sensitivity absent in the model

The figures 3 and 4 present both the human evaluation (four annotators) and the automatic evaluation (using GPT-4o) for the three kinds of data augmentation (lexical, syntactic, and semantic) on the SQuADv2 and PubMedQA datasets (75 QA pairs evaluated on each dataset). We represent in blue the amount of valid generated data, in orange the amount of invalid generated data, and in grey the amount of generated data for which uncertainty was considered. Since human evaluators could assign a confidence score, the standard deviation is also indicated in each box.

On SQuADv2 (figure 3), humans judged 81.5% of items valid versus 77.3% for GPT-4o, a small gap of Δ Valid% $= -4.2$ (negative = model stricter). At the augmentation level, GPT-4o was more lenient on syntactic data $(+9.9)$ but stricter on semantic (-9.9) and lexical (-10.7). Overall, the model broadly tracked human judgments with mild shifts by augmentation type. Confidence remained high (4.8 ± 0.4), suggesting disagreements stem from interpretation rather than uncertainty.

For PubMedQA (figure 4), the pattern reverses, humans judged 66.4% valid, GPT-4o 79.2% ($\Delta = +12.8$). This permissive bias persisted across semantic $(+13.1)$, syntactic $(+13.7)$, and lexical $(+12.8)$. NLP Experts applied stricter criteria, introducing more *invalid* and *unsure* labels not captured

by the model. Confidence dropped to 4.5 ± 0.7, reflecting the difficulty of biomedical texts, where domain-specific terminology and higher stakes reduce annotator certainty (Wang et al., 2021).[3]

Taken together, results reveal a domain-dependent shift. In general data (SQuADv2), humans and GPT-4o converge, with disagreements mostly stylistic. In specialized domains (PubMed), humans adopt stricter thresholds while the model remains permissive, widening the gap. A practical implication is that GPT-4o serves well as a fast first-pass screener, leaving experts to review harder edge cases where consensus is fragile.

6. Energy and Emission Analysis

As tracking carbon is increasingly important both to make the environmental costs of ML visible, recent work urges routine reporting of energy and emissions for ML experiments (Strubell et al., 2019; Schwartz et al., 2020). We track energy use and emissions with CodeCarbon[4] (Courty et al., 2024), an open-source Python library that samples hardware power draw during runtime and converts it into total energy (kWh) equivalent emissions. All fine-tuning experiments were run on a single node equipped with 2× NVIDIA H100 PCIe GPUs and an Intel® Xeon® Gold 5418Y CPU.

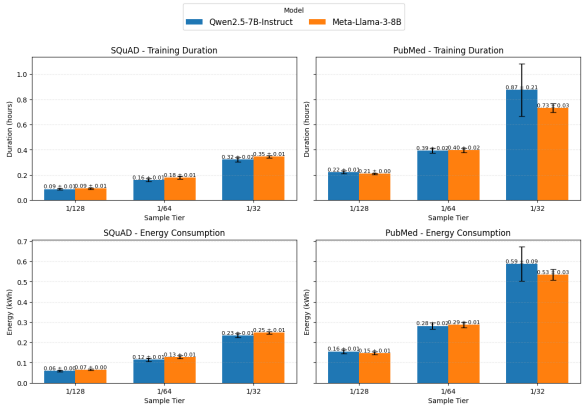


Figure 5: Training time (h) and energy use (kWh) across datasets, sampling tiers, and models. Bars show mean over augmentation types; error bars indicate the standard error of the mean (SEM)

Figure 5 reports training duration (top) and energy consumption (bottom) for SQuADv2 and PubMedQA at three sampling tiers ($\frac{1}{128}$, $\frac{1}{64}$, $\frac{1}{32}$). Results are averaged over the five augmentation conditions (Baseline, Lexical, Syntactic, Semantic, All). Error bars denote standard error, but differences are minimal, indicating that augmentation choice has negligible impact on runtime or energy.

On SQuADv2, Qwen7B-I is consistently faster and slightly more energy-efficient than Llama8B-I, reflecting its smaller parameter count. In contrast, on PubMedQA the trend reverses, with Qwen taking longer and consuming more energy. We hypothesize this stems from differences in tokenization and domain vocabulary. PubMedQA's biomedical terms may yield longer effective sequences for Qwen. This highlights how efficiency can depend not only on parameter count but also on model–data interaction (Hoffmann et al., 2022; Zhou et al., 2024).

7. Conclusion

We systematically compared lexical, syntactic, and semantic data augmentation for extractive QA under low-resource conditions, with fidelity-aware validation via GPT-4o and human audit. Our experiments across SQuADv2 (general) and PubMedQA (biomedical) yield three main findings: (i) semantic augmentation is the only strategy that consistently improves biomedical QA at the smallest scale ($1/128$), while carefully curated baselines often suffice in the general domain; (ii) performance scales non-monotonically with data size, revealing an interaction between LoRA adaptation and supervision volume that warrants careful calibration; and (iii) label prediction can diverge from reasoning quality, underscoring the need for multi-metric evaluation beyond headline accuracy.

Several limitations qualify these conclusions. We evaluated only two model families (Qwen7B-I and Llama8B-I); other architectures such as Mistral-7B may respond differently to the same augmentation types (Liu et al., 2024). Our domain coverage is limited to one general and one biomedical benchmark, legal (Guha et al., 2023) and financial QA (Chen et al., 2021) pose structurally different challenges that future work should address. The GPT-4o validation loop, while efficient, was not calibrated against human preferences, prompt tuning or preference alignment of the judge could further improve augmentation filtering. Finally, we did not scale beyond $1/16$ of the original training sets; larger fractions may reveal additional nonlinearities due to optimizer dynamics or augmentation quality variance. Despite these limitations, we believe this work provides practical guidelines for choosing augmentation strategies under data scarcity and opens the door to more principled, quality-focused augmentation pipelines for domain-adaptable QA.

[3] Confidence values (1–5 scale) are shown inside the bars. For counts below five, values are omitted; in most cases, the score was 5.

[4] https://pypi.org/project/codecarbon/

8. Ethical Considerations

We rely only on public datasets (SQuADv2, PubMedQA) under their respective licenses and do not process personally identifiable information (PII) or protected health information (PHI). All synthetic data are clearly labeled and generated with open models (Llama70B-I for augmentation; Llama8B-I and Qwen7B-I for fine-tuning) to ensure transparency and reproducibility. Human validation was carried out by a small team of expert annotators who participated voluntarily; no demographic or sensitive data were collected. To account for environmental impact, we tracked hardware, training time, and energy consumption using CodeCarbon. Whenever possible, we favored shorter training schedules and smaller models to reduce resource usage while maintaining accuracy.

Code is available at `https://anonymous.4open.science/r/ssl-4CB8`.

9. References

Chris Alberti, Daniel Andor, Emily Pitler, Jacob Devlin, and Michael Collins. 2019. Synthetic QA corpora generation with roundtrip consistency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6168–6173, Florence, Italy. Association for Computational Linguistics.

Vittorio Castelli, Rishav Chakravarti, Saswati Dana, Anthony Ferritto, Radu Florian, Martin Franz, Dinesh Garg, Dinesh Khandelwal, Scott McCarley, Michael McCawley, Mohamed Nasr, Lin Pan, Cezar Pendus, John Pitrelli, Saurabh Pujar, Salim Roukos, Andrzej Sakrajda, Avi Sil, Rosario Uceda-Sosa, Todd Ward, and Rong Zhang. 2020. The TechQA dataset. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1269–1278, Online. Association for Computational Linguistics.

Yung-Chieh Chan, George Pu, Apaar Shanker, Parth Suresh, Penn Jenks, John Heyer, and Sam Denton. 2024. Balancing cost and effectiveness of synthetic data generation strategies for llms. *arXiv preprint arXiv:2409.19759*.

Jie Chen, Yupeng Zhang, Bingning Wang, Wayne Xin Zhao, Ji-Rong Wen, and Weipeng Chen. 2024. Unveiling the flaws: Exploring imperfections in synthetic data and mitigation strategies for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 14855–14865, Miami, Florida, USA. Association for Computational Linguistics.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. 2021. FinQA: A dataset of numerical reasoning over financial data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3697–3711, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Arijit Chowdhury and Aman Chadha. 2024. Generative data augmentation using llms improves distributional robustness in question answering. In *Proceedings of the 18th conference of the European chapter of the association for computational linguistics: student research workshop*, pages 258–265.

Benoit Courty, Victor Schmidt, Sasha Luccioni, Goyal-Kamal, MarionCoutarel, Boris Feld, Jérémy Lecourt, LiamConnell, Amine Saboni, Inimaz, supatomic, Mathilde Léval, Luis Blanche, Alexis Cruveiller, ouminasara, Franklin Zhao, Aditya Joshi, Alexis Bogroff, Hugues de Lavoreille, Niko Laskaris, Edoardo Abati, Douglas Blank, Ziyao Wang, Armin Catovic, Marc Alencon, Michał Stęchły, Christian Bauer, Lucas Otávio N. de Araújo, JPW, and MinervaBooks. 2024. mlco2/codecarbon: v2.4.1.

Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Yihan Cao, Zihao Wu, Lin Zhao, Shaochen Xu, Fang Zeng, Wei Liu, Ninghao Liu, Sheng Li, Dajiang Zhu, Hongmin Cai, Lichao Sun, Quanzheng Li, Dinggang Shen, Tianming Liu, and Xiang Li. 2025. Auggpt: Leveraging chatgpt for text data augmentation. *IEEE Transactions on Big Data*, 11(3):907–918.

Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A survey of data augmentation approaches for NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 968–988, Online. Association for Computational Linguistics.

Dan Friedman and Adji Bousso Dieng. 2023. The vendi score: A diversity evaluation metric for machine learning. *Transactions on Machine Learning Research*.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. 2025. A survey on LLM-as-a-Judge.

Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Ré, Adam Chilton, Aditya Narayana, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel N. Rockmore, Diego Zambrano, Dmitry Talisman, Enam Hoque, Faiz Surani, Frank Fagan, Galit Sarfaty, Gregory M. Dickinson, Haggai Porat, Jason Hegland, Jessica Wu, Joe Nudell, Joel Niklaus, John Nay, Jonathan H. Choi, Kevin Tobia, Margaret Hagan, Megan Ma, Michael Livermore, Nikon Rasumov-Rahe, Nils Holzenberger, Noam Kolt, Peter Henderson, Sean Rehaag, Sharad Goel, Shang Gao, Spencer Williams, Sunny Gandhi, Tom Zur, Varun Iyer, and Zehua Li. 2023. Legalbench: a collaboratively built benchmark for measuring legal reasoning in large language models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Zhen Guo, Peiqi Wang, Yanwei Wang, and Shangdi Yu. 2023. Improving small language models on pubmedqa via generative data augmentation. In *LLM4AI'23: Workshop on Foundations and Applications in Large-scale AI Models -Pre-training, Fine-tuning, and Prompt-based Learning*, Long Beach, CA, USA.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. ArXiv preprint arXiv:2203.15556.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Trans. Inf. Syst.*, 43(2).

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. Technical report, OpenAI.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577, Hong Kong, China. Association for Computational Linguistics.

Qiao Jin, Zheng Yuan, Guangzhi Xiong, Qianlan Yu, Huaiyuan Ying, Chuanqi Tan, Mosha Chen, Songfang Huang, Xiaozhong Liu, and Sheng Yu. 2022. Biomedical question answering: A survey of approaches and challenges. *ACM Comput. Surv.*, 55(2).

Saurav Kadavath, Tom Conerly, Amanda Askell, T. J. Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zachary Dodds, Nova Dassarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, John Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom B. Brown, Jack Clark, Nicholas Joseph, Benjamin Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. 2022. Language models (mostly) know what they know. *ArXiv*, abs/2207.05221.

Adam Tauman Kalai, Ofir Nachum, Santosh S. Vempala, and Edwin Zhang. 2025. Why language models hallucinate. ArXiv preprint 2509.04664v1.

Julien Khlaut, Corentin Dancette, Elodie Ferreres, Alaedine Bennani, Paul Hérent, and Pierre Manceron. 2024. Efficient medical question answering with knowledge-augmented question generation. In *Proceedings of the 6th Clinical Natural Language Processing Workshop*, pages 10–20, Mexico City, Mexico. Association for Computational Linguistics.

Qin Liu, Fei Wang, Nan Xu, Tianyi Lorena Yan, Tao Meng, and Muhao Chen. 2024. Monotonic paraphrasing improves generalization of language model prompting. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 9861–9877, Miami, Florida, USA. Association for Computational Linguistics.

Shayne Longpre, Yi Lu, Zhucheng Tu, and Chris DuBois. 2019. An exploration of data augmentation and sampling techniques for domain-agnostic question answering. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 220–227, Hong Kong, China. Association for Computational Linguistics.

Arindam Mitra, Luciano Del Corro, Guoqing Zheng, Shweti Mahajan, Dany Rouhana, Andres Codas, Yadong Lu, Wei ge Chen, Olga Vrousgos, Corby Rosset, Fillipe Silva, Hamed Khanpour, Yash

Lara, and Ahmed Awadallah. 2024. Agentinstruct: Toward generative teaching with agentic flows.

Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. 2021. Deep double descent: Where bigger models and more data hurt. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124003.

Sania Nayab, Giulio Rossolini, Marco Simoni, Andrea Saracino, Giorgio Buttazzo, Nicolamaria Manes, and Fabrizio Giacomelli. 2025. Concise thoughts: Impact of output length on LLM reasoning and cost. ArXiv preprint arXiv:2407.19825.

Sergey Pletenev, Maria Marina, Daniil Moskovskiy, Vasily Konovalov, Pavel Braslavski, Alexander Panchenko, and Mikhail Salnikov. 2025. How much knowledge can you pack into a lora adapter without harming llm? In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 4309–4322.

Raul Puri, Ryan Spring, Mohammad Shoeybi, Mostofa Patwary, and Bryan Catanzaro. 2020. Training question answering models from synthetic data. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5811–5826, Online. Association for Computational Linguistics.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392. Association for Computational Linguistics.

Revanth Gangi Reddy, Bhavani Iyer, Md Arafat Sultan, Rong Zhang, Avi Sil, Vittorio Castelli, Radu Florian, and Salim Roukos. 2020. End-to-end QA on COVID-19: Domain adaptation with synthetic training. ArXiv preprint arXiv:2012.01414.

Maximilian Schmidt, Andrea Bartezzaghi, and Ngoc Thang Vu. 2024. Prompting-based synthetic data generation for few-shot question answering. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 13168–13178.

Roy Schwartz, Jesse Dodge, Noah A. Smith, and Oren Etzioni. 2020. Green ai. *Commun. ACM*, 63(12):54–63.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Minju Seo, Jinheon Baek, James Thorne, and Sung Ju Hwang. 2024. Retrieval-augmented data augmentation for low-resource domain tasks. ArXiv preprint arXiv:2402.13482v1.

Siamak Shakeri, Cicero Nogueira dos Santos, Henghui Zhu, Patrick Ng, Feng Nan, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. 2020. End-to-end synthetic data generation for domain adaptation of question answering systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5445–5460, Online. Association for Computational Linguistics.

Marco Antonio Sobrevilla Cabezudo, Marcio Lima Inacio, and Thiago Alexandre Salgueiro Pardo. 2024. Investigating paraphrase generation as a data augmentation strategy for low-resource AMR-to-text generation. In *Proceedings of the 17th International Natural Language Generation Conference*, pages 663–675, Tokyo, Japan. Association for Computational Linguistics.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650, Florence, Italy. Association for Computational Linguistics.

Guy Tevet and Jonathan Berant. 2021. Evaluating the evaluation of diversity in natural language generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 326–346, Online. Association for Computational Linguistics.

Kanix Wang, Robert Stevens, Halima Alachram, Yu Li, Larisa Soldatova, Ross King, Sophia Ananiadou, Maolin Li, Fenia Christopoulou, Jose Luis Ambite, Joel Matthew, Sahil Garg, Ulf Hermjakob, Daniel Marcu, Emily Sheng, Tim Beißbarth, Edgar Wingender, Aram Galstyan, and Andrey Rzhetsky. 2021. Nero: a biomedical named-entity (recognition) ontology with a large, annotated corpus reveals meaningful associations

through text embedding. *npj Systems Biology and Applications*, 7.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Joonho Yang, Seunghyun Yoon, Hwan Chang, Byeongjeong Kim, and Hwanhee Lee. 2025. Hallucinate at the last in long response generation: A case study on long document summarization. ArXiv preprint arXiv:2505.15291v2.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. 2018. Qanet: Combining local convolution with global self-attention for reading comprehension. In *Proceedings of ICLR*, Vancouver, Canada.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

Ming Zhong, Yang Liu, Da Yin, Yuning Mao, Yizhu Jiao, Pengfei Liu, Chenguang Zhu, Heng Ji, and Jiawei Han. 2022. Towards a unified multi-dimensional evaluator for text generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2023–2038, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Zixuan Zhou, Xuefei Ning, Ke Hong, Tianyu Fu, Jiaming Xu, Shiyao Li, Yuming Lou, Luning Wang, Zhihang Yuan, Xiuhong Li, Shengen Yan, Guohao Dai, Xiao-Ping Zhang, Yuhan Dong, and Yu Wang. 2024. A survey on efficient inference for large language models. ArXiv preprint arXiv:2404.14294v3.

Yuchang Zhu, Huizhe Zhang, Bingzhe Wu, Jintang Li, Zibin Zheng, Peilin Zhao, Liang Chen, and Yatao Bian. 2025. Measuring diversity in synthetic datasets. In *Forty-second International Conference on Machine Learning*.

A. Augmentation Prompts

Below are the prompts supplied to `Llama-3.1-70B-Instruct` for each augmentation type. Each prompt receives the original context, question, and answer as input, with temperature 0.6 and top-p 0.9.

Lexical augmentation prompt.

```
You are given a context and an
original question-answer pair.
Rephrase the question using
different vocabulary and word
choices while maintaining the
same meaning and answer.

Return the result strictly in
JSON with the following struc-
ture: {"questions": ["..."],
"answers": ["..."]}
```

Syntactic augmentation prompt.

```
You are given a context and an
original question-answer pair.
Rewrite the question using dif-
ferent grammatical structure
or sentence construction while
preserving the same meaning and
answer.

Return the result strictly in
JSON with the following struc-
ture: {"questions": ["..."],
"answers": ["..."]}
```

Semantic augmentation prompt.

```
You are given a context and an
original question-answer pair.
Create exactly 1 new question-
short answer pair that asks about
a different facet of the context
(not a paraphrase). You need to
be creative.

Return the result strictly in
JSON with the following struc-
ture: {"questions": ["..."],
"answers": ["..."]}
```

B. Hyperparameters

Hyperparameter	Value
LoRA rank (r)	16
LoRA α	16
LoRA dropout	0.05
Target modules	q/k/v/o/gate/up/down_proj
Optimizer	AdamW (fused)
Learning rate	1×10^{-4}
LR scheduler	Cosine
Warmup ratio	0.1
Weight decay	0.01
Max gradient norm	0.8
Batch size (per device)	32
Gradient accumulation	4
Epochs	4
Max sequence length	2048
Precision	bf16 + tf32
Seed	42

Table 5: Full hyperparameters used for LoRA fine-tuning

Introducing corpora Hlava Cor and Hlava AD: Human Label Variation in Coreference and Discourse Relations

Anna Nedoluzhko, Šárka Zikánová, Jiří Mírovský, Milan Straka and Eva Hajičová
Charles University, Faculty of Mathematics and Physics, Institute of Formal and Applied Linguistics
Prague, Czech Republic
{nedoluzhko, zikanova, mirovsky, straka, hajicova}@ufal.mff.cuni.cz

Abstract

As previous research on annotator disagreement in discourse phenomena has shown, understanding text coherence varies considerably from one individual to another. To explore this phenomenon, we created two corpora with multiple annotations of Czech texts, accompanied by annotators' explanations of their choices. The first corpus consists of 1,024 contexts annotated in parallel by three annotators. It captures differences in the identification of coreference across various text types and grammatical-semantic categories, including pronouns, full noun phrases, and anaphoric adverbials. The second corpus comprises 512 contexts, annotated in parallel by five annotators, and focuses on identifying discourse relations in attributive and non-attributive constructions. Both corpora achieve a comparable inter-annotator agreement of approximately 60–65%. For coreference annotation, agreement tends to be lower in cases where automatic coreference resolution models disagree, suggesting that when the models disagree, the examples tend to be more difficult or ambiguous for human annotators to interpret. The annotators' comments, both for coreference and discourse relations, further reveal differences in interpretation, varying levels of confidence in text understanding, and individual reading strategies.

Keywords: comprehension of coherence, multiple annotations, commented annotation

1. Introduction

During our long-term development of corpora focused on discourse relations, coreference, and information structure, we observed that certain linguistic phenomena tend to produce lower inter-annotator agreement than others. This observation aligns with a growing body of research on annotation evaluation taking into account different perspectives (Basile et al., 2021) and Human Label Variation (HLV, Plank, 2022), which suggests that disagreement in NLP is often a reflection of inherent linguistic complexity rather than mere noise. It is possible to identify several factors that appear to contribute to this variation, ranging from the polysemy of synsemantic signals to word order patterns (see, e.g., Zikánová, 2024). Furthermore, annotator-related factors, such as the subjective perception of the relative importance of text segments, play a significant role. Such interpretative diversity is well-documented in large-scale projects; for instance, Poesio et al. (2020) and Recasens et al. (2010) have consistently observed that ambiguity in textual meaning naturally reduces agreement in coreference tasks.

In the present paper, we introduce two corpora created to investigate human label variation. **Hlava Cor** (Human Label Variation in Coreference (Nedoluzhko et al., 2026)) contains annotations of coreference and focuses on human label variation related to genericity, subjectivity and underspecification. **Hlava AD** (Human Label Variation in Attribution and Discourse (Šárka Zikánová et al., 2024)) comprises annotations of discourse relations in attributive and non-attributive constructions.

In developing these corpora, we took into account various hypotheses about the possible causes of annotation disagreement, based on our previous annotation experience and analyses. For the coreference corpus, we specifically distinguish generic and specific expressions, as well as different grammatical–semantic categories, including pronouns, full noun phrases, and anaphoric adverbials. We further hypothesize that coreference resolution models may partially predict cases of underspecification, in the sense that where models disagree, human annotators are also likely to find the interpretation of coreference relations problematic (see Section 4.3 and 4.4).

For discourse relations, we assume that in texts with larger segments of direct or indirect speech, it may be more difficult for recipients to recognize which of the upcoming sentences are still related to the direct/indirect speech (see Section 5).

We argue that multiple annotation, and especially the detailed comments accompanying each annotation, provide rich material for investigating the reasons and nuances of human label variation, enhancing our understanding of how people interpret texts differently. We understand our corpora as datasets providing many types of commented structures, and thus serving as a base for future psycholinguistic experiments dealing with special cases. Furthermore, these corpora may serve as reference datasets for evaluating the reliability of previously single-annotated corpora.

Proceedings of the Workshop on Structured Linguistic Data and Evaluation, pages 237–247

2. Related Work

The interpretation of linguistic meaning is never entirely fixed or uniform; it is shaped by context, perspective, and the inherent ambiguity of natural language. This fundamental indeterminacy influences all levels of linguistic analysis, including discourse and coreference annotation, where human judgments often diverge even under detailed annotation guidelines. Researchers in discourse annotation address this challenge in different ways. For example, Marchal et al. (2022) discuss methods evaluating annotation quality across multiple annotators. Building on the "ecologically valid" explanations introduced by Jiang et al. (2023), recent work by Weber-Genzel et al. (2024) utilizes a two-round annotation procedure where participants justify their label choices through natural language explanations. In these studies, annotators classify the relations between clauses as entailment, contradiction, or neutral and provide commentaries that allows researchers to distinguish between interpretative variation and annotation errors. Other studies emphasize the importance of explicitly capturing interpretative plurality (Plank, 2022, Basile et al., 2021, Crible et al., 2019).

New corpora are published capturing multiple understandings of discourse relations, e.g. in the RST approach (Peng et al., 2022, Polakova et al., 2024, Hewett and Stede, 2025), or implicit discourse relations in the PDTB approach (Scholman et al., 2022, Yung et al., 2024). None of these datasets include annotators' comments on their choices.

Comparative research on spoken and written data further demonstrates that modality contributes to interpretative diversity: spoken language tends to involve more implicit and multifunctional relations (Rehbein et al., 2016, Cuenca, 2017, Crible et al., 2019), though its annotation consistency is not necessarily lower than that of written texts (Zufferey and Crible, 2015).

Ambiguity in textual meaning naturally reduces inter-annotator agreement (IAA), as has been consistently observed in large-scale coreference annotation projects (Weischedel et al., 2011; Zeldes, 2017; Recasens and Martí, 2010; Poesio, 2020). While analyses of annotation disagreement have occasionally been included (Recasens et al., 2011; Pradhan et al., 2012), such studies are typically limited to small, task-specific subsets of data (cf. Levine and Zeldes, 2025). A few corpora explicitly encode ambiguous or near-identity cases (Uryupina et al., 2020; Bourgonje and Stede, 2020; Ogrodniczuk et al., 2013), but these remain statistically rare, leaving much of the natural variation in human interpretation unexplored.

Poesio et al. (2019) present a preliminary analysis of disagreements in a corpus of anaphoric information in 542 English documents crowdsourced through a game-with-a-purpose. The corpus contains multiple concurrent annotations of about 108 thousand markables, with 20 judgments on average per markable (12 annotations and 8 validations). Expert analysis of a sample of the data shows, however, that genuine ambiguity occurs only in about 9% of the markables and the other cases of annotators' disagreement should be accounted to annotators' errors and to various limitations of the coding scheme and the annotation interface.

In our analysis, we use the Prague Dependency Treebank - Consolidated 2.0 (PDT-C 2.0; (Hajič et al., 2024)) as the primary data source for extracting and then multiply annotating the data.

Namely, we use its subcorpus Prague Dependency Treebank (PDT) for the written mode and Prague Dependency Treebank of Spoken Czech (PDTSC) for the spoken mode. The PDT represents texts from the Czech newspapers from 1990's, while the PDTSC contains transcripts of conversations between interviewers and Holocaust survivors or contemporary witnesses of communism. The PDT-C 2.0 corpus contains manual annotations of coreference and discourse relations, along with multiple linguistic layers ranging from morphology to tectogrammatics. Annotation in PDT-C 2.0 follows the principle of a single correct decision, with approximately 10% of the data annotated in parallel to assess inter-annotator agreement (IAA). The collection comprises both spoken and written subcorpora, enabling comparative analyses of discourse phenomena across modalities.

3. Hlava Cor and Hlava AD: General Settings

To study human label variation, we extracted texts primarily from different parts of the PDT-C 2.0 (see detailed description in Sections 4.3 and 5) and created Hlava Cor and Hlava AD. The overall statistics for the annotated data are listed in Table 1. Hlava Cor contains 1,024 cases annotated by three annotators, while Hlava AD contains 512 cases annotated by five annotators.[1]

Both corpora consist of short Czech text segments containing the relevant linguistic phenomena. Each text segment was multiply annotated by several annotators. The annotations were performed on the linear texts presented in an Excel spreadsheet.

[1]The discrepancy in the size of the two corpora and the number of participating annotators does not stem from a specific theoretical or methodological requirement, but rather reflects the successive stages of their development and the varying availability of resources during each phase of the project.

Table 1: Overall Statistics for Hlava Cor and Hlava AD

Corpus Name	Number of Cases	Number of Annotators
Hlava Cor	1,024	3
Hlava AD	512	5

The annotation team consisted of native Czech speakers, primarily philology students (aged 18–30) with a basic linguistic background but no formal training in specific theoretical frameworks. This lack of prior exposure was intentional to ensure that text comprehension was not biased by theoretical presuppositions. While a core group of five annotators worked on Hlava AD, the Hlava Cor corpus was annotated by six individuals organized into two groups by three annotators. This larger group included the same five participants from the Hlava AD task plus one additional annotator.

The annotation process is designed not only to capture the final annotation decisions but also to elicit the annotators' interpretative considerations. For this reason, all annotations obligatorily include annotators' comments explaining their choices which are crucial for our investigation.

4. Hlava Cor: Human Language Variation in Coreference

The annotation task for Hlava Cor was defined as the identification of coreference, i.e. reference to the same extra-linguistic entity, concept, or situation.

4.1. Hlava Cor: Annotation Process

Annotators were instructed to read the column `Sentence` with a highlighted expression in question and the left-hand context in columns `Adjacent Context` and `Distant Context` (see Table 2)[2] and to find a potential antecedent in the preceding context if any. No potential antecedents have been pre-annotated.[3] Each annotation output contains (i) the antecedent expression if it appeared in the preceding context, (ii) a numeric value indicating the degree to which context was required for interpretation (0 = no distant context needed, 1 = distant context necessary, 2 = even wider context needed), and (iii) a free-text comment explaining the annotator's reasoning or uncertainty.

[2]The column `Distant Context` is not shown in the example for space reasons.

[3]In Table 2, two antecedents are highlighted in bold because the table shows an already completed annotation. The bold highlighting is used only to make the example easier to read in this paper; the annotators did not see these highlights during the annotation process.

The annotation guidelines specified detailed conventions for the form and syntactic scope of antecedents. Annotators were required to record the full nominal phrase as it appears in the text, excluding prepositions unless they constitute part of a named entity or are internal to the referring expression. Dependent modifiers were to be included, while relative clauses were to be excluded. Deviations from these rules had to be explicitly documented in the comment field. Further specifications addressed the treatment of references to clauses or larger textual segments, possessive and other adjectival expressions, coordinated and discontinuous antecedents. When no suitable antecedent was found, annotators entered NE ("no antecedent"), and in cases of complete indecision they recorded a question mark.

4.2. Hlava Cor: Annotation Example

An example of the annotated segment is presented in Table 2. For the expression *vozidlo* 'vehicle' in the column `Sentence`, the annotators were asked to identify an antecedent (if any) and record it in the `ANN_ante` column. In the `ANN_comment` column, the annotators were asked to justify their decision or provide any additional remarks.

This example illustrates the annotation by two annotators out of three who arrived at different conclusions. Annotator 1 (ANN1) selected *vůz* 'car' as the antecedent of *vozidlu* 'vehicle' in `Sentence`, arguing that the expression does not refer to the specifically described model driven by the journalist, but rather to the vehicle in general. Annotator 2 (ANN2), on the other hand, chose *vozu* 'car (genitive)' as the antecedent, noting that both expressions refer to the same entity.

For space reasons, the annotation of the third annotator is not shown; this annotator made the same decision as ANN1 but expressed more uncertainty in the comment. We also omit the annotation of the need for context, which was zero for all three annotators in this case.

4.3. Hlava Cor: Data Description and Statistics

The corpus consists of 1,024 cases, each marked by three annotators. The segments for annotation were selected according to several dimensions of data division. First, we took the same number of spoken and written text segments. Most of the text segments have been excerpted from the PDT-C 2.0, a few segments come from the Czech Academic Corpus 2.0 (CAC) (Hladká et al., 2008). For the written data, we additionally used examples from

Table 2: Example of coreference annotation by two annotators

Adjacent Context	Sentence	ANN1 ante	ANN1 comment	ANN2 ante	ANN2 comment
Firma Renault Česká republika nám k redakčnímu testu poskytla verzi s turbodieselovým motorem s označením RT. Sice jsme v úvodu uvedli, komu lze **vůz** [ANN1] především doporučit. Snad nám nebudou mít případní zájemci z kategorie 'mladá rodina s více dětmi' za zlé, když jim neprozradíme, kde vzít potřebných 969820 korun. V případě testovaného **vozu** [ANN2] dovybaveného dvěma střešními okny, autorádiem s ovládáním pod volantem a ve vlnové metalíze pak ještě přes 66 tisíc korun navrch.	Věnujeme se však raději samotnému **vozidlu**, neboť svezení s ním opravdu stojí za to.	vůz	Neodkazuje na detailně popsaný, konkrétní model vozu, s nímž zrovna jel novinář, ale obecně na vozidlo.	vozu	Oba výrazy označují stejnou skutečnost.

Adjacent Context: The company Renault Czech Republic provided us with a version featuring a turbo diesel engine, designated RT, for an editorial test. Although we mentioned at the beginning who this **car** [ANN1] can primarily be recommended to, we hope that potential buyers from the category of "young families with several children" will not hold it against us if we do not reveal where to get the required 969,820 crowns. In the case of the tested **car** [ANN2], additionally equipped with two sunroofs, a car radio with steering-wheel controls, and metallic paint, the price increases by more than 66,000 crowns.

Sentence: However, let us turn to the **vehicle** itself, since driving it is really worth the experience.
ANN1 ante: *vůz* 'car'
ANN1 comment: Does not refer to the specifically described model that the journalist was driving, but rather to the vehicle in general.
ANN2 ante: *vozu* 'car'(genitive)
ANN2 comment: Both expressions refer to the same entity.

the archive of iRozhlas.[4] Table 3 shows the data distribution from these resources.

Table 3: Structure of Hlava Cor: written vs. spoken modes

Mode	Source Corpus	Number of Cases
Spoken	PDT-C 2.0	505
	Czech Academic Corpus 2.0 (CAC)	7
	Total Spoken	512
Written	PDT-C 2.0	461
	iRozhlas	40
	Czech Academic Corpus 2.0 (CAC)	11
	Total Written	512
Grand Total		**1,024**

We also considered the referential status of nominal groups for all categories, and divided the segments according to the specific and generic reference of the nominal groups and pronouns in coreferential relations (see Table 4). The division of the specific/generic reference did not apply for the cases with the local pronominal adverb *tam* 'there'.

Table 4: Structure of Hlava Cor: specific vs. generic reference

Reference Category	Number of Cases
full NPs + to 'it' with specific reference	384
full NPs + to 'it' with generic reference	384
tam 'there' (division is not relevant)	256
Total Cases	**1,024**

Another dimension was the division of our data into coreference types where the anaphoric expression is a full nominal group (noun, noun + adjective, etc.), pronominal coreference with *to* 'it', or local coreference with the pronominal adverb *tam* 'there' (see Table 5).

Table 5: Structure of Hlava Cor: nominal and pronominal anaphors

Anaphor Type	Number of Cases
full NPs	512
pronominal coreference with to 'it'	256
local coreference with tam 'there'	256
Total Segments	**1,024**

Being interested in exploring multiple readings of coreference, we aimed to extract examples that exhibit ambiguous or otherwise vague instances of coreference, and at the same time to allow for comparison with cases where coreference appears to be easily identifiable. In an effort to assess such a distinction in potential examples automatically, we have first passed the examples to five different Transformer-based coreference resolution models.[5] All potential examples were pre-selected to be theoretically ambiguous, i.e., based on morphological properties, at least two possible antecedents were present within a five-sentence context, making

[4] the internet archive of the Czech news outlet iRozhlas, https://www.irozhlas.cz/zpravy-archiv

[5] We trained 14 models on CorefUD 1.2 (Popel et al., 2024) using the coreference resolution system CorPipe (Straka, 2024), the winner of the CRAC 2024 shared task (Novák et al., 2024), and selected 5 achieving the best results on the two Czech CorefUD datasets.

disagreement possible. Based on the output of the five models, the potential examples were divided into two groups: (i) examples where all five models agreed on the antecedent, and (ii) examples where at least two models disagreed, with preference for examples with larger number of disagreements. Final examples for annotation were selected from these two groups and included in Hlava Cor in a 1:2 ratio. The ratio was uneven intentionally, as we assumed that cases of models' disagreement would provide more informative material for analysis. Table 6 presents the data distribution according to the models' agreement and disagreement.

Table 6: Structure of Hlava Cor: model agreement vs. model disagreement

Coreference Status	Number of Cases
model agreement	352
model disagreement	672
Total Cases	**1,024**

Table 7 shows how various categories and dimensions in Tables 3 to 6 relate to each other. For example, the first line states that there are 128 cases with an NP type of the anaphor in spoken data with generic coreference (in 84 of them the coreference resolvers disagreed, in 44 cases they were in agreement). The features can be combined to larger groups of equal size, for example 'to' spoken (128 cases) vs. 'to' written (128 cases), or 'tam' (256 cases) vs. 'to' (256 cases), or spoken (512 cases) vs. written (512 cases).

Table 7: Numbers of cases for detailed combinations of features in Hlava Cor

Feature Combination	Models Disagreed	Models Agreed	Total
NP spoken GEN	84	44	128
NP spoken SPEC	84	44	128
NP written GEN	84	44	128
NP written SPEC	84	44	128
tam spoken	84	44	128
tam written	84	44	128
to spoken GEN	42	22	64
to spoken SPEC	42	22	64
to written GEN	42	22	64
to written SPEC	42	22	64
Total	**672**	**352**	**1,024**

4.4. Hlava Cor: Inter-Annotator Agreement

For each category described in Section 4.3 (and summarized in Tables 3, 4, 5, and 6) we calculate the inter-annotator agreement of our three human annotators for setting the coreferential antecedent.[6] Table 8 presents the IAA results for coreference annotation in Hlava Cor. Overall, the agreement of all three annotators reached 49%, while partial agreement (at least two out of three annotators) reached 83%. The average pairwise agreement for Hlava Cor reaches 60.4%.[7]

Table 8: Hlava Cor: Inter-Annotator Agreement Overview

Category	All 3	At Least 2
Text Mode (from Table 3)		
written data	45%	81%
spoken data	53%	86%
Referential Status (from Table 4)		
generic coreference (GEN)	45%	82%
specific coreference (SPEC)	48%	82%
Grammatical Form (from Table 5)		
full NPs	51%	86%
pronominal (to 'it')	38%	75%
local (tam 'there')	57%	88%
Model Status (from Table 6)		
model agreement	66%	91%
model disagreement	39%	79%
All data	**49%**	**83%**

The inter-annotator agreement results reflect the inherent complexity of coreference annotation, particularly in naturally occurring data and in cases involving vague or underspecified referential relations. When broken down by referential status, agreement for generic coreference (45%) and specific coreference (48%) is relatively similar, suggesting that both types of reference pose comparable challenges for human annotation. This contradicts our initial observations and is possibly related to the lexical composition of generic nominal phrases in Hlava Cor. Regarding grammatical form, the pronominal coreference with *to* 'it' showed the lowest (38%) agreement as compared to full nominal groups (51%) and the local adverbial *tam* 'there' (57%), confirming that non-nominal and underspecified *to* 'it' may be more ambiguous. A more detailed analysis of these patterns will be the subject of further research.

Another relevant observation supports our hypothesis that instances where the models dis-

[6] Exact match was required. In order to avoid typo-related errors, the annotators were asked to copy/paste the segments from the original text; also, the disagreements were manually checked by an expert to find and fix cases of insignificant omissions (for example, a comma at the end of the coppied text, a missing character etc.).

[7] This number is not included in Table 8.

agreed tend to be more difficult for human annotators as well. Human annotation on the model-disagreement subset reached only 39% inter-annotator agreement (IAA), whereas cases where the models agreed showed considerably higher consistency, with an IAA of 66%. Notably, an IAA of 66% is still relatively low, especially given that the models themselves were in agreement. This fact clearly deserves special attention and can be partially explained by the inherent implicitness and ambiguity of coreference relations in our data. Model agreement in such cases likely reflects a combination of chance and biases induced by the training data. Overall, these findings suggest that model disagreement may serve as an indicator of interpretative complexity in the data.

Taken from a slightly different perspective, we can also examine how many distinct annotation choices were made for a single case, see Table 9. The table presents the absolute numbers of unique coreference choices per context annotated by the three annotators. The specific reasons for inter-annotator disagreement are not analyzed in detail here; however, in general, they often stem from differences in the level of detail or interpretation of the referential scope of the entities.

Table 9: Hlava Cor: Distribution of Coreference Choices by Annotator Agreement (absolute numbers).

Choice Type	Agreement Level	Cases
One choice	Full (3/3)	503
Two different antecedent	Partial (2/3)	351
Three different antecedents	No agreement (1/3)	170
Total Annotated Choices		**1,024**

5. Hlava AD: Human Label Variation in Attribution and Discourse

The corpus contains 512 short Czech text segments with explicit inter-sentential discourse relations, multiply annotated by five annotators. All texts in Hlava AD come from PDT-C 2.0.

Similarly to Hlava Cor, we separately considered the written and spoken modalities.

Based on our previous findings of frequent cases of inter-annotator disagreement on discourse relations (Zikánová, 2024), we hypothesized that a higher measure of disagreement in inter-sentential relations might be related to the presence of attributive constructions (sentences with verbs of thinking or saying). The supposed reason is that it may be difficult for recipients to recognize whether the upcoming sentence is related to the governing clause (author's speech) or to the dependent direct or indirect speech.

Table 10: Structure of the Hlava AD (number of items)

	Spoken data	Written data	**Total**
Attribution	96	125	**221**
Other	173	118	**291**
Total	**269**	**243**	**512**

To analyze the influence of attributive constructions on interpretive variation, we divided the Hlava AD data into two subsets: one containing attributive constructions with verbs of thinking and saying (including both direct and indirect speech) and a control subset with non-attributive constructions. Table 10 summarizes the structure of Hlava AD concerning the spoken/written and attribution/non-attribution dimensions.

The text segments were chosen from the original source (PDT-C 2.0) based on the syntactic and discourse structure. We searched for explicit inter-sentential discourse relations with a complex sentence in the role of the left argument. This complex sentence contained at least one dependent clause and its governing verb was either a verb of thinking or saying (attributive constructions) or not (non-attributive constructions). In non-attributive items, we filtered out all the text segments containing verbs of thinking or saying in any other role.

5.1. Hlava AD: Annotation Process

The annotators were shown pairs of sentences which had been related by an explicit discourse relation in the previous annotation of PDT-C 2.0. The `Right sentence` column contains a discourse connective (marked green, see Figure 1). The `Left sentence` column consists of several clauses; in the attributive group of items, the left sentence contains a governing verb of saying / thinking, whereas in the control group, another semantic type of verb is used instead. For better understanding of the text, annotators were additionally given the three preceding sentences in the `Preceding context` column.

The annotators' initial task was to identify the clause in the left sentence, to which the discourse connective relates the right sentence (they searched for the so-called "target of the relation").

To technically simplify the task for the annotators, we pre-marked the finite verbs (or parts of verbal forms) in clauses in the `Left sentence` in red, see Figure 1. Thus, the verbs served as identifiers of the whole clauses; annotators entered the chosen verb into the corresponding column.

In case of coordination between two verbal forms (clauses), the entire construction could be annotated as the target of the relation. Coordination was

Figure 1: Annotators' prompt in the Hlava AD annotation task

Preceding context	Left sentence	Right sentence
Španělská sekce prestižní Asociace evropských novinářů pozvala již po šesté různorodé spektrum středoevropských politiků, publicistů a filosofů, aby početným posluchačům letního universitního semináře s názvem Střední Evropa mezi Bruselem a Moskvou přednesli své úvahy o tom, co se uprostřed kontinentu děje. Pozornost od počátku budila polská účast. Vítěz nedávných voleb Aleksander Kwasniewski a premiérka poražené vlády Hanna Suchocká, každý ze svého úhlu, ale stejně přesvědčivě popsali, co se stalo v Polsku.	Kwasniewski opakovaně **zdůraznil**, že z cesty zásadních proměn země **nelze** sejít: pravice a levice se budou **přít** se středem o tempo změn, **ale** o základní vzorec reforem **není** sporu.	Co **však** je vážné, je nevelký zájem veřejnosti o věci veřejné.

[Preceding context: The Spanish section of the prestigious Association of European Journalists has invited for the sixth time a diverse spectrum of Central European politicians, publicists and philosophers to present their reflections on what is happening in the middle of the continent to the numerous audience of the summer university seminar entitled Central Europe between Brussels and Moscow. The Polish participation attracted attention from the beginning. The winner of the recent elections, Aleksander Kwasniewski, and the Prime Minister of the defeated government, Hanna Suchocka, each from their own perspective, but equally convincingly described what happened in Poland.

Left sentence: Kwasniewski has repeatedly **emphasized** that the country **cannot** be diverted from the path of fundamental changes: the right and left will **argue** with the center about the pace of change, **but** there **is** no dispute about the basic formula of reforms.

Right sentence: What is serious, **however**, is the public's lack of interest in public affairs.]

identified by the pre-marked (blue) conjunctions or punctuation marks.

Thus, the annotators could mark red verbal forms or blue conjunctive phrases/marks as targets of the relation expressed by the green discourse connective. Additionally, they had a possibility to use an exclamation mark (!) to indicate that the target was in the preceding or far left context, and they used a question mark (?) to signal that they could not find any target in the given text (e.g., they did not understand the text segment as a whole).

The second annotation task was to mark to what degree the annotator needed the previous context (before the left sentence) to understand the discourse relation expressed by the discourse connective. There were three possible answers: 0 (I did not need the previous context); 1 (I needed the previous context to understand and it helped me); and 2 (I needed the previous context to understand, but it did not help me).

The third annotation task required annotators to explain their choice of the target of the relation by entering comments in a separate plain text field.

5.2. Hlava AD: Annotation Example

Figure 1 presents an example with an attributive construction in the left sentence (*Kwasniewski **emphasized** that...*). Annotators interpret this segment in two different ways. Some annotators identify the main clause with the governing verb *emphasized* as the target of the discourse relation expressed by the discourse connective *however*. Their comments explain that Kwasniewski emphasized political happenings. On the other hand, the annotators stress that what is truly important is what is going on among ordinary people. This group of annotators does not see the right sentence as a part of the Kwasniewski's speech.

Conversely, other annotators mark the clause *that the country **cannot** be diverted...* as the target. They claim in their comments that Kwasniewski knows about the upcoming changes and is concerned about the public's lack of interest in public affairs. Thus, they understand the right sentence as a part of the Kwasniewski's speech.

5.3. Hlava AD: Inter-Annotator Agreement

We calculate the inter-annotator agreement on setting the target of a discoure relation, expressed by the discourse connective in the right sentence. We can obtain theoretically 1-5 solutions from 5 annotators; nevertheless, some syntactic structures in the left sentences are simpler, they do not provide up to 5 potential targets of the discourse relation. On the other hand, the disagreement can be increased by the possibility to use the mark "?" (not understandable) and "!" (relation to a distant left

Table 11: IAA on the target of a discourse relation in Hlava AD (percentage points)

	Agreement (1 solution from 5 annotators)	Disagreement (2-5 solutions from 5 annotators)
In general	38%	62%
Simple structures (2 possible targets)	48%	52%
Complex structures (3-10 pos. targets)	27%	73%

context). The average pairwise agreement disregarding the syntactic complexity reached 64.9%. The results for the IAA regarding the syntactic complexity of the left sentence (and, subsequently, a different number of potential targets) is presented in Table 11.

A more detailed analysis of the IAA on the target of a discourse relation can be found in Zikánová et al. (2025), where the multiple annotation serves as a test dataset for the default golden data. According to the authors, the measure of the IAA in Hlava AD is strongly affected by the syntactic complexity of the left sentence; however, it is influenced neither by the presence of the attribution, nor by the text mode (spoken or written).

6. Observations across Hlava Cor and Hlava AD

Given that Hlava Cor covers a dataset twice the size of Hlava AD, and that Hlava AD was annotated by five annotators while Hlava Cor involved only three parallel annotations, the results are not fully comparable. However, it is possible to compare the average pairwise agreement on target identification, which neutralizes the difference in the number of annotators. For Hlava AD, the average pairwise agreement reaches 64.9%, while for Hlava Cor it is 60.4%.

In both corpora, annotators marked the need for context (see Sections 4.1 and 5.1). The analysis of the annotations reveals substantial variation across annotators, while the overall tendencies remain similar in both corpora. What we observe likely reflects, to some extent, the annotators' degree of confidence in their interpretation of the text, as well as individual differences in annotation style. Some annotators tend to consult a broader preceding context to verify their interpretation, while others consider their immediate reading sufficient without referring to additional context. To a certain degree, this also mirrors the annotator's thoroughness and attention to detail.

Particular attention should be paid to the annotators' comments, which, in our view, represent the most interesting part of our annotation. Comments were mandatory in both corpora, and they certainly deserve a separate study beyond the scope of this paper. They capture the annotators' doubts about their chosen decisions, indicate possible alternative interpretations, and often describe in detail the strategies underlying their choices. Especially intriguing is the overlap between the annotators' comments and their decisions in other tasks. In some cases, the antecedents were chosen identically (inter-annotator agreement), yet the comments still mentioned alternative options. Conversely, in certain cases of inter-annotator disagreement, the analysis shows that the annotators reasoned in a very similar way but ultimately arrived at different decisions.

7. Conclusions

In this paper, we presented and described two datasets created to study human label variation: Hlava Cor (Nedoluzhko et al., 2026), focused on coreference, and Hlava AD (Šárka Zikánová et al., 2024), focused on attribution and discourse relations. Hlava Cor explores coreference with respect to the reference status of anaphoric expressions (specific or generic), their form (pronouns, full noun phrases, and anaphoric adverbials), and the extent to which coreference resolution models agree on identifying these relations. Hlava AD pays special attention to attribution and non-attributive constructions. Both corpora include spoken and written data and consider both registers equally.

The introduced corpora offer space for a more detailed analysis of label variation in human annotation. Preliminary observations suggest individual differences between annotators and the presence of personal annotation styles. Of particular interest are the annotators' comments, which often explain annotation choices, reflect subjective judgments, and point to interpretative ambiguity. These and related aspects will be the focus of our future work based on the data obtained from the corpora.

Acknowledgements

The authors gratefully acknowledge support from the Grant Agency of Czech Republic, project 24-11132S, as well as the OP JAK project CZ.02.01.01/00/23_020/0008518 of the Ministry of Education, Youth and Sports of the Czech Republic.

The research reported in the present contribution has been using language resources developed, stored and distributed by the LINDAT/CLARIAH-CZ project of the Ministry of Education, Youth and Sports of the Czech Republic (LM2023062).

8. Bibliographical References

V. Basile, B. Plank, D. Hovy, P. Van Der Lee, L. Van Der Plas, and M. Poesio. 2021. We need to consider disagreement in evaluation. In *Proceedings of 1st Workshop on Benchmarking, ACL*, pages 15–21.

Peter Bourgonje and Manfred Stede. 2020. The Potsdam commentary corpus 2.2: Extending annotations for shallow discourse parsing. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1061–1066, Marseille, France. European Language Resources Association.

L. Crible, M. J. Cuenca, L. Degand, C. Fuentes, and J. Vandelanotte. 2019. Functions and translations of underspecified discourse markers in TED talks: A parallel corpus study on five languages. *Journal of Pragmatics*, 142:139–155.

Maria Josep Cuenca. 2017. Discourse markers in speech: Distinctive features and corpus annotation. *Dialogue & Discourse*.

Freya Hewett and Manfred Stede. 2025. Disagreements in analyses of rhetorical text structure: A new dataset and first analyses. In *Proceedings of the 19th Linguistic Annotation Workshop (LAW-XIX-2025)*, pages 35–47, Vienna, Austria. Association for Computational Linguistics.

Nan-Jiang Jiang, Chenhao Tan, and Marie-Catherine de Marneffe. 2023. Ecologically valid explanations for label variation in NLI. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10622–10633, Singapore. Association for Computational Linguistics.

Lauren Levine and Amir Zeldes. 2025. Subjectivity in the annotation of bridging anaphora. In *Proceedings of the 19th Linguistic Annotation Workshop (LAW-XIX-2025)*, pages 48–59, Vienna, Austria. Association for Computational Linguistics.

M. Marchal, B. Hladká, V. Lojda, and J. Jírka. 2022. Establishing annotation quality in multi-label annotations. In *Proceedings of COLING*, pages 3659–3668.

Michal Novák, Barbora Dohnalová, Miloslav Konopik, Anna Nedoluzhko, Martin Popel, Ondrej Prazak, Jakub Sido, Milan Straka, Zdeněk Žabokrtský, and Daniel Zeman. 2024. Findings of the third shared task on multilingual coreference resolution. In *Proceedings of the Seventh Workshop on Computational Models of Reference, Anaphora and Coreference*, pages 78–96, Miami. Association for Computational Linguistics.

Maciej Ogrodniczuk, Katarzyna Glowinska, Maciej Kopec, Agata Savary, and Magda Zawisławska. 2013. Polish coreference corpus. In *Human Language Technology. Challenges for Computer Science and Linguistics - 6th Language and Technology Conference, LTC 2013, Poznan, Poland, December 7-9, 2013. Revised Selected Papers*, volume 9561 of *Lecture Notes in Computer Science*, pages 215–226. Springer.

Siyao Peng, Yang Janet Liu, and Amir Zeldes. 2022. GCDT: A Chinese RST treebank for multigenre and multilingual discourse parsing. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 382–391, Online only. Association for Computational Linguistics.

Barbara Plank. 2022. The "problem" of human label variation: On ground truth in data, modeling and evaluation. In *Proceedings of EMNLP*, pages 10671–10682.

Massimo Poesio. 2020. Ambiguity. In D. Gutzmann, L. Matthewson, C. Meier, H. Rullmann, and T. Zimmermann, editors, *The Wiley Blackwell Companion to Semantics*. Wiley.

Massimo Poesio, Jon Chamberlain, Silviu Paun, Juntao Yu, Alexandra Uma, and Udo Kruschwitz. 2019. A crowdsourced corpus of multiple judgments and disagreement on anaphoric interpretation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1778–1789, Minneapolis, Minnesota. Association for Computational Linguistics.

Lucie Polakova, Jiří Mírovský, Šárka Zikánová, and Eva Hajicova. 2024. Developing a Rhetorical Structure Theory treebank for Czech. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 4802–4810, Torino, Italia. ELRA and ICCL.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Proceedings of the Sixteenth Conference on Computational Natural Language Learning (CoNLL 2012)*, Jeju, Korea.

Marta Recasens, Eduard Hovy, and M. Antònia Martí. 2011. Identity, non-identity, and near-

identity: Addressing the complexity of coreference. *Lingua*, 121:1138–1152.

Marta Recasens and M. Antònia Martí. 2010. AnCora-CO: Coreferentially annotated corpora for spanish and catalan. *Language Resources and Evaluation*, 44:315–345.

I. Rehbein, M. Reznicek, R. Schüller, H. Schüppert, and M. Stede. 2016. Annotating discourse relations in spoken language: A comparison of the PDTB and CCR frameworks. In *Proceedings of LREC'16*, pages 1039–1046, Portorož.

Merel Scholman, Tianai Dong, Frances Yung, and Vera Demberg. 2022. DiscoGeM: A crowdsourced corpus of genre-mixed implicit discourse relations. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3281–3290, Marseille, France. European Language Resources Association.

Milan Straka. 2024. CorPipe at CRAC 2024: Predicting zero mentions from raw text. In *Proceedings of the Seventh Workshop on Computational Models of Reference, Anaphora and Coreference*, pages 97–106, Miami. Association for Computational Linguistics.

Olga Uryupina, Ron Artstein, Andrea Bristot, Federico Cavicchio, Fabio Delogu, Kenia J. Rodriguez, and Massimo Poesio. 2020. Annotating a broad range of anaphoric phenomena, in a variety of genres: the ARRAU corpus. *Natural Language Engineering*, 26:95–128.

Leon Weber-Genzel, Siyao Peng, Marie-Catherine De Marneffe, and Barbara Plank. 2024. VariErr NLI: Separating annotation error from human label variation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2256–2269, Bangkok, Thailand. Association for Computational Linguistics.

Ralph Weischedel, Eduard Hovy, Mitchell Marcus, Martha Palmer, Robert Belvin, Sameer Pradhan, Lance Ramshaw, and Nianwen Xue. 2011. Ontonotes: A large training corpus for enhanced processing. In *Handbook of Natural Language Processing and Machine Translation: DARPA Global Autonomous Language Exploitation*, pages 54–63. Springer-Verlag, New York.

Frances Yung, Merel Scholman, Sarka Zikanova, and Vera Demberg. 2024. DiscoGeM 2.0: A parallel corpus of English, German, French and Czech implicit discourse relations. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 4940–4956, Torino, Italia. ELRA and ICCL.

Amir Zeldes. 2017. The GUM corpus: Creating multilayer resources in the classroom. *Language Resources and Evaluation*, 51:581–612.

Šárka Zikánová. 2024. Text structure and its ambiguities: Corpus annotation as a helpful guide. In *Proceedings of the 24th Conference Information Technologies – Applications and Theory (ITAT 2024)*, pages 2–12, Košice, Slovakia. Univerzita Pavla Jozefa Šafárika v Košiciach, Košice, Slovakia, CEUR-WS.org.

Šárka Zikánová, Anna Nedoluzhko, Jiří Mírovský, and Eva Hajičová. 2025. Gold data and multiple understanding of discourse relations. In *28th International Conference on Text, Speech and Dialogue (Part II)*, volume 16030 of *Lecture Notes in Computer Science*, pages 250–262, Cham, Switzerland. Friedrich-Alexander-Universität Erlangen/Nürnberg, Springer.

Sandrine Zufferey and Ludivine Crible. 2015. Using a unified taxonomy to annotate discourse markers in speech and writing. In *Proceedings of the 11th Joint ACL - ISO Workshop on Interoperable Semantic Annotation*, London.

9. Language Resource References

Jan Hajič, Eduard Bejček, Alevtina Bémová, Eva Buráňová, Eva Fučíková, Eva Hajičová, Jiří Havelka, Jaroslava Hlaváčová, Petr Homola, Pavel Ircing, Jiří Kárník, Václava Kettnerová, Natalia Klyueva, Veronika Kolářová, Lucie Kučová, Markéta Lopatková, David Mareček, Marie Mikulová, Jiří Mírovský, Anna Nedoluzhko, Michal Novák, Petr Pajas, Jarmila Panevová, Nino Peterek, Lucie Poláková, Martin Popel, Jan Popelka, Jan Rompoltl, Magdaléna Rysová, Jiří Semecký, Petr Sgall, Johanka Spoustová, Milan Straka, Pavel Straňák, Pavlína Synková, Magda Ševčíková, Jana Šindlerová, Jan Štěpánek, Barbora Štěpánková, Josef Toman, Zdeňka Urešová, Barbora Vidová Hladká, Daniel Zeman, Šárka Zikánová, and Zdeněk Žabokrtský. 2024. Prague Dependency Treebank - Consolidated 2.0 (PDT-C 2.0).

Barbora Hladká, Jan Hajič, Jirka Hana, Jaroslava Hlaváčová, Jiří Mírovský, and Jan Raab. 2008. The czech academic corpus 2.0 guide. *The Prague Bulletin of Mathematical Linguistics*, 89:41–96.

Anna Nedoluzhko, Jiří Mírovský, Šárka Zikánová, Eva Hajičová, Bianca Chuffartová, Šárka

Dohnalová, Lucie Hartmanová, Eliška Nodlová, Dominik Teska, and Františka Zikánová. 2026. Human Label Variation in Coreference (Hlava Cor). Data/software, ÚFAL MFF UK, Prague, Czech Republic, LINDAT/CLARIAH-CZ: http://hdl.handle.net/11234/1-6131.

Martin Popel, Michal Novák, Zdeněk Žabokrtský, Daniel Zeman, Anna Nedoluzhko, Kutay Acar, David Bamman, Peter Bourgonje, Silvie Cinková, Hanne Eckhoff, Gülşen Cebiroğlu Eryiğit, Jan Hajič, Christian Hardmeier, Dag Haug, Tollef Jørgensen, Andre Kåsen, Pauline Krielke, Frédéric Landragin, Ekaterina Lapshinova-Koltunski, Petter Mæhlum, M. Antònia Martí, Marie Mikulová, Anders Nøklestad, Maciej Ogrodniczuk, Lilja Øvrelid, Tuğba Pamay Arslan, Marta Recasens, Per Erik Solberg, Manfred Stede, Milan Straka, Daniel Swanson, Svetlana Toldova, Noémi Vadász, Erik Velldal, Veronika Vincze, Amir Zeldes, and Voldemaras Žitkus. 2024. Coreference in universal dependencies 1.2 (CorefUD 1.2). LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL).

Šárka Zikánová, Jiří Mírovský, Anna Nedoluzhko, Eva Hajičová, Šárka Dohnalová, Anna Kmječová, Eliška Nodlová, and Dominik Teska. 2024. Human Label Variation in Attribution and Discourse (Hlava AD). Data/software, ÚFAL MFF UK, Prague, Czech Republic, LINDAT/CLARIAH-CZ: http://hdl.handle.net/11234/1-5819.

Author Index