LREC-COLING 2024

The 5th International Workshop on Designing Meaning Representation (DMR 2024) @LREC-COLING-2024

Workshop Proceedings

Editors
Claire Bonial, Julia Bonn, and Jena D. Hwang

21 May, 2024
Torino, Italia

Proceedings of the Fifth International Workshop on Designing Meaning Representations (DMR 2024) @LREC-COLING-2024

ISBN 978-2-493814-39-5
ISSN 2951-2093 (COLING); 2522-2686 (LREC)

Jointly organized by the ELRA Language Resources Association
and the International Committee on Computational Linguistics

Preface

In her 2023 ACL Lifetime Achievement Award acceptance speech, Dr. Martha Palmer (University of Colorado, Boulder) sums up her 50 years of research in AI and NLP in six words: “Finding meaning, quite literally, in words.” Now in its fifth iteration, the International Designing Meaning Representations Workshop brings together researchers from around the world who endeavor to do the same.

While deep learning methods have led to many breakthroughs in practical natural language applications, most notably in Machine Translation, Machine Reading, Question Answering, Recognizing Textual Entailment, and so on, there is still a sense among many NLP researchers that we have a long way to go before we can develop systems that can actually "understand" human language and explain the decisions they make. Indeed, “understanding” natural language entails many different human-like capabilities, and they include but are not limited to the ability to track entities in a text, understand the relations between these entities, track events and their participants, understand how events unfold in time, and distinguish events that have actually happened from events that are planned or intended, are uncertain, or did not happen at all. “Understanding” also entails human-like ability to perform qualitative and quantitative reasoning, possibly with knowledge acquired about the real world. We believe a critical step in achieving natural language understanding is to design meaning representations for text that have the necessary meaning “ingredients” that help us achieve these capabilities.

These proceedings showcase the work of researchers who are producers and consumers of meaning representations, who come together in this forum every year to gain a deeper understanding of the key elements of meaning that are the most valuable to the NLP community. The workshop provides an opportunity for meaning representation researchers to examine critically existing frameworks with the goal of using their findings to inform the design of next-generation meaning representations. Together, we explore opportunities and identify challenges in the design and use of meaning representations in multilingual settings, and seek to understand the relationship between distributed meaning representations (trained on large data sets using network models) versus symbolic meaning representations (carefully designed and annotated by CL researchers).

This year’s Designing Meaning Representation workshop honors Dr. Palmer’s 50-year research journey with a special theme on resources, approaches, and applications that draw upon her manifold contributions to the field: Treebanks, PropBanks, VerbNets, OntoNotes, Abstract Meaning Representation (AMR), and Uniform Meaning Representation (UMR). These resources share attention to semantic detail combined with scalability and, therefore, an ability to generalize to and support a variety of different NLP applications and tasks. Indeed, the applicability of Dr. Palmer’s research extends beyond the textual to the multimodal, where she has broadly contributed to cross-modal event understanding. Thus, DMR 2024 highlights the depth and the breadth of Dr. Palmer’s contributions and their influence over the field of natural language processing by including original works that have leveraged, expanded, or been inspired by the “Marthaverse of Meaning.” With gratitude, we recognize Dr. Palmer’s long tenure of dedication to outstanding mentorship that has been so powerful for the many students who have gone on to shape the NLP research community and the field at large.

These proceedings include papers presented at the 5th Designing Meaning Representation workshop on May 21, 2024, held in conjunction with the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024) in Torino, Italy. DMR 2024 received 25 submissions, out of which 17 papers have been accepted to be presented at the workshop as talks (six papers) and posters or virtual short presentations

(11 papers). The papers address topics ranging from meaning representation methodologies to issues in meaning representation parsing, to the adaptation of meaning representations to specific applications and domains, to cross-linguistic issues in meaning representation. In addition to the oral paper presentations and poster session, DMR 2024 also featured invited talks by Fei Xia (University of Washington) and Owen Rambow (Stony Brook University), entitled "If Data Could Talk" and "Propositional Content and Commitment to Truth," respectively.

Message from the Workshop Chairs

We thank our organizing committee for its continuing organization of the DMR workshops, and the LREC-COLING 2024 workshop chairs for their support. We are grateful to all of the authors for submitting their papers to the workshop and our program committee members for their dedication and their thoughtful reviews. We thank our invited speakers for making the workshop a uniquely valuable discussion of linguistic annotation research. Finally, we thank Martha Palmer for her continued innovation, inspiration, motivation, and encouragement in our research community and in our own lives.

Claire Bonial, Julia Bonn, and Jena D. Hwang

Message from Special Honoree Martha Palmer

First, I want to thank all the workshop chairs, Claire Bonial, Julia Bonn, Jena Hwang, and the organizing committee, Lucia Donatelli, Jan Hajič, Alexis Palmer, Nathan Schneider, Nianwen Xue, for this very delightful tribute. They made me cry and I am eternally grateful to them, for their kindness to me and for their friendship.

I've led a charmed life. I've gotten to work on these incredibly interesting and challenging problems, with all of these amazing people, like Jim Martin at CU. What more could anyone ask for? Every work environment, there have been intelligent, dedicated, curious folks - students, postdocs, staff and faculty - and together we were able to try to solve the insoluble problems, to dream the impossible dreams. It doesn't get better than that.

Over the years I've realized there is something special about people who are fascinated by semantics. They are in awe of the mystery of language, and how it is that we manage to communicate using it.

Knowledge is fostered by curiosity; wisdom is fostered by awe (Heschel, A, 1965).[1]

Folks who work on semantics have to have a certain humility, an ability to recognize that they are never completely sure if they have exactly the right answer. Semantics often defies being cleanly defined. A large part of this has to do with the ability to be comfortable with uncertainty. Things that are clearly right or wrong, black or white, are so much easier to deal with. Yet so much of the world and of life, like semantics, doesn't fit those categories neatly. Perhaps, if we

[1] Complete Heschel quote:

> *Knowledge is fostered by curiosity; wisdom is fostered by awe. Awe precedes faith; it is the root of faith. We must be guided by awe to be worthy of faith. Forfeit your sense of awe, let your conceit diminish your ability to revere, and the universe becomes a market place for you. The loss of awe is the avoidance of insight. A return to reverence is the first prerequisite for a revival of wisdom, for the discovery of the world as an allusion to God.*

Abraham J. Heschel, Who Is Man? (Stanford, CA: Stanford University Press, 1965), 88–89, isbn=9780804702669 study semantics long enough with a sufficient quota of awe, we will all gain a tiny modicum of wisdom. Maybe it is no accident that almost every collaborative project, every student supervision, has turned into a life-long friendship. It is a privilege to be able to associate with such exceptional people.

I will end with a quote from a Center for Action and Contemplation meditation from February 24, 2024, by Kate Bowler.

> *"I had a very tender podcast conversation with theologian and ethicist Stanley Hauerwas. We have worked together for almost two decades now, and I rely on him to be incredibly certain about what makes a life good and virtuous. . . . After describing how many twists and turns that life had taken, he had come to a conclusion: "The ability to live well is the ability to live without so many certainties."*

We are all living life well, aren't we?

Martha Stone Palmer

Workshop Chairs

Claire Bonial, Army Research Labs
Julia Bonn, University of Colorado Boulder
Jena D. Hwang, Allen Institute for AI

Organizing Committee

Lucia Donatelli, Vrije Universiteit Amsterdam
Jan Hajič, Charles University
Alexis Palmer, University of Colorado Boulder
Nathan Schneider, Georgetown University
Nianwen Xue, Brandeis University

Program Committee

Omri Abend, Hebrew University of Jerusalem
Zahra Azin, Carleton University
Katrien Beuls, Université de Namur, Belgium
Abhidip Bhattacharyya, University of Massachusetts Amherst
Johan Bos, University of Groningen
Chloé Braud, CNRS - IRIT
Alastair Butler, Hirosaki University, Japan
Katie Conger, University of Colorado, Boulder
Valeria de Paiva, Topos Institute, Berkeley, CA
Katrin Erk, University of Texas, Austin
Kilian Evang, Heinrich Heine University, Düsseldorf
Federico Fancellu, 3M HIS
Frank Ferraro, University of Maryland, Baltimore County
Anette Frank, University of Heidelberg
Annemarie Friedrich, University of Augsburg
Kira Griffit, UPenn/LDC
Udo Hahn, JULIE Lab, FSU Jena & TexKnowlogy
Daniel Hershcovich, University of Copenhagen
Julia Hockenmaier, University of Illinois
Nancy Ide, Vassar College and Brandeis University
Elisabetta Jezek, University of Pavia
Paul Landes, University of Illinois at Chicago
Alex Lascarides, University of Edinburgh
Bin Li, Nanjing Normal University China
Yunyao Li, Apple
Adam Meyers, New York University
Sarah Moeller, University of Florida
Philippe Muller, University of Toulouse
Skatje Myers, University of Wisconsin Madison
Joakim Nivre, Uppsala Universitet and RISE
Juri Opitz, Heidelberg University
Miriam R. L. Petruck, FrameNet
Alain POLGUÈRE, Université de Lorraine, CNRS, ATILF
James Pustejovsky, Brandeis University
Weiguang Qu, Nanjing Normal University
Michael Regan, University of Washington
Djamé Seddah, Inria Paris

Manfred Stede, University of Potsdam
Kevin Stowe, Educational Testing Service
Harish Tayyar Madabushi, The University of Bath
Zdeňka Urešová, MFF UK, Charles University, Prague
Ashwini Vaidya, IIT Delhi
Paul Van Eecke, Vrije Universiteit Brussel
Clare Voss, ARL
Shira Wein, Georgetown University
Susan Windisch Brown, University of Colorado
Kristin Wright Bettner, University of Colorado Boulder
Hongzhi Xu, Shanghai International Studies University
Annie Zaenen, Stanford University
Deniz Zeyrek, Middle East Technical University
Heike Zinsmeister, Universität Hamburg

Publicity Chair

Kristine Stenzel, University of Colorado Boulder

Invited Speakers

Owen Rambow, Stony Brook University
Fei Xia, University of Washington

Table of Contents

Workshop Program

May 21, 2024

9:00–10:30	**Morning Session 1: Intro, Keynote 1, and in-person paper talk**
9:00–9:10	***Intro***
9:10–10:10	*Keynote 1: If Data Could Talk* Fei Xia
10:10–10:30	*PropBank-Powered Data Creation: Utilizing Sense-Role Labelling to Generate Disaster Scenario Data* Mollie Frances Shichman, Claire Bonial, Taylor A. Hudson, Austin Blodgett, Francis Ferraro and Rachel Rudinger
10:30–11:00	**Coffee Break**
11:00–13:00	**Morning Session 2: in-person paper talk, lightning talks, and special honoree activities**
11:00–11:20	*Aspect Variability and the Annotation of Aspect in the IMAGACT Ontology of Action* Massimo Moneglia and Rossella Varvara
11:20–11:40	*NoVRol: A semantic role lexicon of Norwegian verbs* Henrik Torgersen, Erlend Ø. Ravnanger, Lars Hellan and Dag Haug

May 21, 2024 (continued)

11:40–12:08 **Virtual Lightning Talks**

11:40–11:48 *Expanding Russian PropBank: Challenges and Insights for Developing New SRL Resources*
Skatje Myers, Roman Khamov, Adam Pollins, Rebekah Tozier, Olga Babko-Malaya and Martha Palmer

11:50–11:58 *Unveiling Semantic Information in Sentence Embeddings*
Leixin Zhang, David Burian, Vojtěch John and Ondřej Bojar

12:00–12:08 *A Quantum Theory of Terms and New Challenges to Meaning Representation of Quanterms*
Diego Burgos

12:10–13:00 **Special Session: Honoring Martha Palmer**

12:10–12:25 *A special message from our Guest of Honor*
Martha Palmer

12:25–13:00 ***Marthaverse activity***

13:00–14:00 **Lunch Break**

14:00–16:00 **Afternoon Session 1: poster session and keynote 2**

May 21, 2024 (continued)

14:00–15:00 **Poster Session**

VOLARE - Visual Ontological LAnguage REpresentation
Werner Winiwarter

YARN is All You Knit: Encoding Multiple Semantic Phenomena with Layers
Siyana Pavlova, Maxime Amblard and Bruno Guillaume

Argument Sharing in Meaning Representation Parsing
Maja Buljan, Stephan Oepen and Lilja Øvrelid

Mapping PropBank Argument Labels to Czech Verbal Valency
Jan Hajič, Eva Fučíková, Marketa Lopatkova and Zdeňka Urešová

Lexicalized Meaning Representation (LMR)
Jorge Baptista, Sónia Reis, João Dias and Pedro Santos

Adjudicating LLMs as PropBank Adjudicators
Julia Bonn, Harish Tayyar Madabushi, Jena D. Hwang and Claire Bonial

Extending VerbNet's Verb-Specific Features to Enhance Selectional Preferences of Semantic Roles
Susan Windisch Brown

Chinese UMR annotation: Can LLMs help?
Haibo Sun, Nianwen Xue, Jin Zhao, Liulu Yue, Yao Sun, Keer Xu and Jiawei Wu

15:00–16:00 *Keynote 2: Propositional Content and Commitment to Truth*
Owen Rambow

May 21, 2024 (continued)

16:00–16:30 Coffee Break

16:30–18:00 Afternoon Session 2: in-person paper talks

16:30–16:50 *Accelerating UMR Adoption: Neuro-Symbolic Conversion from AMR-to-UMR with Low Supervision*
Claire Benet Post, Marie C. McGregor, Maria Leonor Pacheco and Alexis Palmer

16:50–17:10 *The Relative Clauses AMR Parsers Hate Most*
Xiulin Yang and Nathan Schneider

17:10–17:30 *Gaining More Insight into Neural Semantic Parsing with Challenging Benchmarks*
Xiao Zhang, Chunliu Wang, Rik van Noord and Johan Bos

17:30–18:00 *Outro*
Claire Bonial, Julia Bonn, Jena D. Hwang

PropBank-Powered Data Creation: Utilizing Sense-Role Labelling to Generate Disaster Scenario Data

Mollie Shichman[1], **Claire Bonial**[2], **Taylor Hudson**[3],
Austin Blodgett[2], **Francis Ferraro**[4], **Rachel Rudinger**[1]
[1]University of Maryland College Park, [2]Army Research Lab,
[3]Oak Ridge Applied Universities, [4] University of Maryland Baltimore County
{mshich, rudinger}@umd.edu, claire.n.bonial.civ@army.mil

Abstract

For human-robot dialogue in a search-and-rescue scenario, a strong knowledge of the conditions and objects a robot will face is essential for effective interpretation of natural language instructions. In order to utilize the power of large language models without overwhelming the limited storage capacity of a robot, we propose PropBank-Powered Data Creation. PropBank-Powered Data Creation is an expert-in-the-loop data generation pipeline which creates training data for disaster-specific language models. We leverage semantic role labeling and Rich Event Ontology resources to efficiently develop seed sentences for fine-tuning a smaller, targeted model that could operate onboard a robot for disaster relief. We developed 32 sentence templates, which we used to make 2 seed datasets of 175 instructions for earthquake search and rescue and train derailment response. We further leverage our seed datasets as evaluation data to test our baseline fine-tuned models.

Keywords: PropBank, Object Affordances, Synthetic Data Creation, Fine-tuning

1. Introduction

In dangerous and dynamic problem spaces like search and rescue, instructing a robot agent in the field via natural language offers a flexible means of communication with a low cognitive burden on rescue workers. However, it is imperative that the robot agent be able to correctly understand and execute natural language instructions from its human operator. For example, for the instruction “move past the chair and try to find an entrance,” the robot agent should be able to determine if the instruction is related to navigation, interacting with objects with a mechanical arm, identifying obstacles in its environment, or a combination of those options. These instructions are often specific to the disaster scenario in question, the tools required for search and rescue for the given disaster, and the overall environment where the disaster occurred. Finally, the robot agent needs physical common-sense reasoning to effectively follow instructions in such a precarious environment.

Large language models (LLMs) have shown great promise for encoding world knowledge (Petroni et al., 2019), as well as strong performance on instruction following tasks (Ouyang et al., 2022; Wang et al., 2022; Chung et al., 2022). However, these models have drawbacks for human-robot interaction in disaster relief. Instruction LLMs are often unspecialized, aimed at accomplishing a plethora of diverse written tasks rather than specializing in a domain-specific task with its own assumptions and peculiarities. Additionally, LLMs are trained on tasks that do not require a strong basis in physical common sense, including the potential usages of objects, which we term ’affordances.’ As a LLM may not have any specific semantic training, it is unclear how they will perform on relevant semantic scenarios like reasoning about properties of objects. Another challenge is that LLM’s reasoning can be difficult to interpret and predict.

Furthermore, there is a pragmatic limitation of available hardware in robot systems. As LLMs vastly increase in size, it becomes more difficult for smaller hardware systems to use these models. Most robots use one commercially available GPU, and assuming the GPU has 24 GB of memory and the LLM is using 4-bit quantization (Dettmers et al., 2023), the robot could realistically only run an LLM with 40B parameters. A robot working in disaster relief needs many other systems onboard, so memory space is even further limited down to smaller 7 billion or 13 billion parameter models. These smaller models would need fine-tuning to be competent in the field due to their size. However, fine-tuning data for specific types of disasters are not easily available.

We hypothesize a solution to this problem space is to fine-tune small LLMs with a wide variety of disaster-specific data. These LLMs should be able to answer both multiple choice and open ended questions about how to execute different subtasks of the disaster. They should be able to reason about the various objects a robot could come across during a disaster relief mission. This includes knowing the functions of different objects, the different states an object can be in, the relative size and shape of objects, etc. Yet another important task is recogniz-

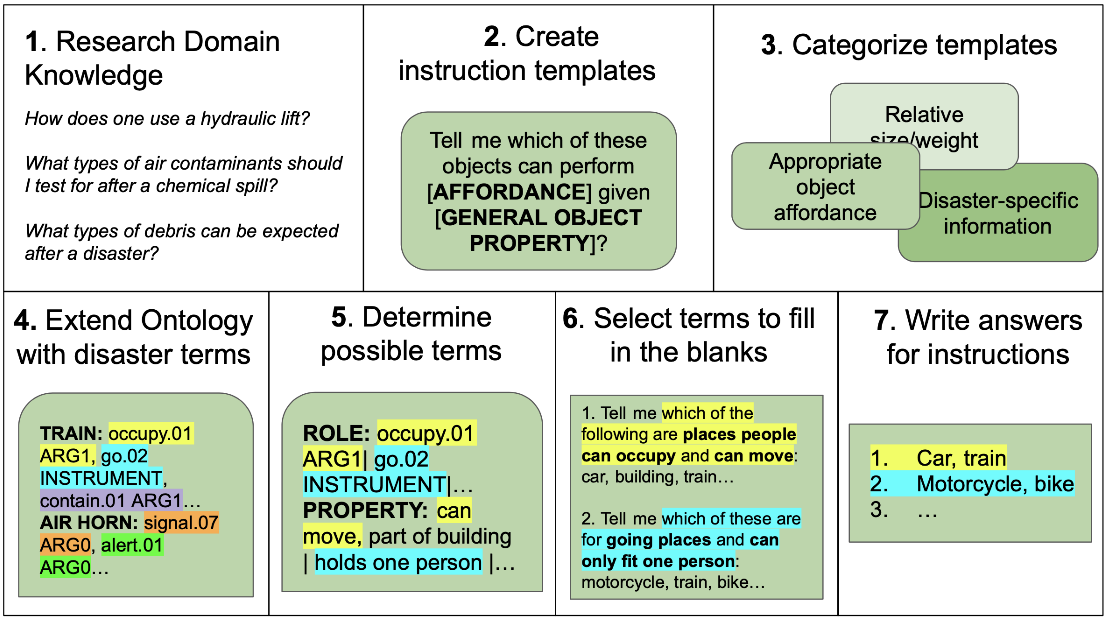

1. Research Domain Knowledge
How does one use a hydraulic lift?
What types of air contaminants should I test for after a chemical spill?
What types of debris can be expected after a disaster?
2. Create instruction templates
Tell me which of these objects can perform [AFFORDANCE] given [GENERAL OBJECT PROPERTY]?
3. Categorize templates
Relative size/weight
Appropriate object affordance
Disaster-specific information
4. Extend Ontology with disaster terms
TRAIN: occupy.01 ARG1, go.02 INSTRUMENT, contain.01 ARG1…
AIR HORN: signal.07 ARG0, alert.01 ARG0…
5. Determine possible terms
ROLE: occupy.01 ARG1| go.02 INSTRUMENT|…
PROPERTY: can move, part of building | holds one person |…
6. Select terms to fill in the blanks
1. Tell me which of the following are places people can occupy and can move: car, building, train…
2. Tell me which of these are for going places and can only fit one person: motorcycle, train, bike…
7. Write answers for instructions
1. Car, train
2. Motorcycle, bike
3. …

Figure 1: The workflow for generating gold-standard instructions. After collecting domain knowledge about different types of questions to be answered, we created templates for the different types of instructions and categorized them to ensure a relatively even distribution of queried knowledge in our results. We then determine the terms, roles, and/or vocabulary that could fill in the templates. Creating these templates allowed us to quickly generate gold standard instructions for object affordances and earthquake search and rescue. These instructions were then used for both perturbing the embeddings of a language model during the training data generation stage and evaluating the resulting fine-tuned model. Instructions corresponding to the occupy.01 ARG1 role are highlighted in yellow. Instructions corresponding to the go.02 INSTRUMENT role are highlighted in blue. More in-depth examples of seed sentences can be found in table 1.

ing what objects have the potential to be dangerous. All of these functionalities are necessary in order for successful human-robot interaction in these disaster scenarios, both for ease of interaction and for the robot agent's successful execution of the instruction. The goal of this work is to create a framework for generating data that can provide a basis for reasoning about this wide variety of tasks. This process can be seen in Figure 1.

While the tasks we want an LLM to accomplish are diverse and ambitious, Taori et al. (2023) has had great success with a similar task to ours. They instruction fine-tuned the LLaMa 7B model to have similar instruction following performance to GPT3.5, a much larger LLM. To do this, they expertly crafted seed instructions that were fed into OpenAI's `text-davinci-003` as In-Context Learning (ICL) for generating high-quality synthetic data (Dong et al., 2023). While effective, their methodology for creating seed sentences for synthetic data generation is not appropriate for our use case for two reasons. For one thing, the seed instructions used by Taori et al. (2023) were created by a group of experts whose broad domain and lack of time constraints meant they could generate uniquely formatted seed instructions on a relatively ad hoc basis. We need a systematized pipeline to ensure that our sentences are generated quickly as well as accurately, and that all relevant areas of our disaster domain are covered by our seed sentences. This is so a robot agent can be deployed quickly and with high accuracy for disasters that place time constraints on when relief efforts must happen. Additionally, the seed instructions were not based in any particular semantics that Taori et al. (2023) wanted their model to "understand", while we need our model to have semantic understanding of the disaster and the objects a robot agent could encounter while navigating it.

To solve these issues, we propose an expert-in-the-loop data generation pipeline called PropBank-Powered Data Creation, which can be seen in Figure 1. In this pipeline, seed sentences are informed by disaster expert knowledge, then created by a linguistic expert in one work day. These seed sentences are then used as in-context learning for synthetic data generation to produce a much larger dataset than would otherwise be possible with a tight timeframe and a highly specialized domain. The seed sentences are constructed using templates rooted in the semantic properties of disaster-relevant senses from the PropBank lexicon (Palmer et al., 2005). These seed instructions also serve as a semantically informed evaluation, since they are not included in the resulting synthetic dataset.

The contributions of this paper are as follows:

1. A process where linguists, with minimal disaster expert input, can quickly generate gold-standard seed sentences to be used during synthetic data generation. This includes 35 sentence templates for generating seed sentences.

2. An ontology of over 300 disaster relevant vocabulary terms that are annotated with PropBank sense-role labels representing the objects' affordances and change of state potentials

3. Two sets of 175 seed sentences: one focused on earthquakes, and one focused on the Ohio Train Derailment[1]

2. Background

In the sections to follow, we provide background information on the source of common-sense object affordance knowledge that we leverage to seed the generation of fine-tuning data, followed by the fine-tuning procedure we adopt.

2.1. Object Properties

As interaction with objects is a major component of the instructions a robot may be given, it is important to have a framework for describing different types of objects and what affordances, or functionalities, a given object may have, as well as the canonical changes of state the object may undergo.

We leverage the Affordance Ontology of disaster-relevant vocabulary terms (Shichman et al., 2023) that adopts a PropBank-style (Palmer et al., 2005) representation of the vocabulary's function and state changes in terms of semantic roles each term played with respect to an event. This resource, an extension of the Rich Event Ontology (Bonial et al., 2021), is a hub mapping event concepts from different semantic role labeling resources and includes "qualia relations," and specifically "telic" relations that denote the affordances of objects in terms of events (Kazeminejad et al., 2018).

The Rich Event Ontology previously only represented a limited number of telic qualia relations expressed between objects and particular events. The Affordance Ontology extends the vocabulary and representations of the Rich Event Ontology by representing object affordances in terms of PropBank sense-role pairings for given senses of events. For example, within the Affordance Ontology, the affordance of a bucket is labeled as an ARG0, or "container" of a *contain.01* event, defined loosely as "hold inside."[2] A box would not only be represented with this same containing affordance, but would also be characterized by a representation of a canonical change of state: to be open (ARG1 of *open.01*) or closed (ARG1 of *close.01*).

The Affordance Ontology provides a basis of a vocabulary of objects that are likely to be present in generic search and rescue scenarios. This means that this resource can serve as a gold-standard set of object properties within our disaster use cases. In this research, we not only use the Affordance Ontology, but also extend it to new objects and affordances leveraging our PropBank-Powered Data Creation workflow (described in detail in section 3).

There are other resources for defining object functionality that we considered for our application—notably the Suggested Upper Merged Ontology (SUMO) (Niles and Pease, 2003), which includes axioms and object definitions to indicate object affordances. However, we preferred to use PropBank because of its elegance in representing the object's functionality and because of the amount of data supporting its approach. Furthermore, SUMO is more focused on connecting semantic concepts stored on the word level rather than fully describing events. Using PropBank, specifically the PropBank rolesets, also allows for our work to be integrated with other Natural Language Understanding resources like Abstract Meaning Representation, which shares the same roleset representation of events (Banarescu et al., 2013) and can distill instructions into action primitives and their corresponding parameters (Bonial et al., 2020).

2.2. Generating Natural Language Instructions

Obtaining high quality language for training and fine-tuning language models is expensive and time consuming. With the rise and improvement of LLMs, significant work is being done to examine if LLMs can do this work with more speed and with the same level of accuracy as crowd-sourcing.

Notably, Wang et al. (2022) developed a framework for prompting a language model to create a diverse set of instructions which could be used to fine-tune said language model. Specifically, the process begins with writing 175 unique seed instructions, then prompting GPT3 to generate a new set of diverse instructions, then filtering out instructions of insufficient quality via ROUGE-L score. Af-

[1] https://www.reuters.com/world/us/ohio-carry-out-controlled-release-chemicals-train-derailment-site-2023-02-06/

[2] `https://propbank.github.io/v3.4.0/frames/contain.html`

ter generating approximately 52,000 instructions, these instructions were then fed back into GPT3 for fine-tuning. This resulted in SELF-INSTRUCT, a fine-tuned GPT3 model that humans rated significantly better on instruction tasks than vanilla GPT3. Furthermore, though it performed worse than all versions of InstructGPT, it was close and still competitive, and required much less human labor (Wang et al., 2022).

Inspired by the success of Wang et al. (2022) and the release of LLaMa (Touvron et al., 2023), Taori et al. (2023) created their own fine-tuned instruction-following model, Alpaca. Alpaca largely followed the same algorithm for generating their own instructions as SELF-INSTRUCT. The major innovation of Alpaca was that it used the output of GPT3 to fine-tune the smaller LLaMa 7B model rather than GPT3 itself. This provided a major performance boost, with humans rating the Alpaca answer to be the preferred one just as often as Vanilla GPT3. We follow the approach of Alpaca, but make use of PropBank to quickly develop seed instructions.

3. PropBank-Powered Data Creation Methodology

To quickly turn expert knowledge from both written and oral sources into a disaster-specific LLM, we aim to develop an efficient way of generating a set of gold standard seed instructions. These seed sentences will then be used as in-context learning for synthetic data generation, which in turn will be used to fine-tune a smaller LLM to enhance its performance on a specific disaster domain.

To create the initial set of seed sentences, we developed the PropBank-Powered Data Creation Pipeline, which relies upon sentence templates with slots that are populated largely by object vocabulary from the Affordance Ontology (Shichman et al., 2023). The vocabulary that can be used within a particular slot is constrained by the PropBank-style representation of properties such as its affordances and change of state potentials. For example, to create a seed sentence querying relative weight, one would take the template “Which of these objects is the lightest? [LIST OF OBJECTS]” and fill in the “blank” with a list of objects that were randomly generated, then refined to only include objects with differentiable weights. Template examples can be seen in Table 1. More complex and elaborate examples can be found in Table 2.

Thus, templates can be semi-automatically populated based on linguistic properties of the template slot, instead of having disaster experts develop dozens of unique instructions. This decreases time to robot deployment while maintaining the accuracy of the seed sentences. The challenge therefore becomes how to effectively template important properties for downstream use.

3.1. Creating the Templates

To tackle the challenge of creating templates for generating seed sentences, we developed an annotation workflow in which graduate student linguistic annotators brainstormed a variety of instructions and questions that a disaster-relief specialist might want a robot to be able to execute or answer. The annotators were instructed not to write instructions outside of a LLM’s capabilities, like image identification or referring to a 3D space the LLM cannot perceive (e.g. "Get that can from your right"). Some examples of brainstormed questions include “What can be used for travel and carry large loads?” and “How can an adult reach the ceiling?”.

The linguistic annotators then moved from the hypothetical to real data by incorporating disaster expert knowledge. For the purposes of this paper, our ‘expert knowledge’ came from written documents about the response to the Ohio train derailment (Air Sampling; Water Sampling; Soil Sampling; Derailment Tools; Yan et al., 2023) and the search and rescue process after earthquakes (Arranz et al., 2023; Hydraulic Rescue Tools; Scarbury, 2015; Thermal Cameras). A separate author collected the expert knowledge, and our annotators reviewed these data before constructing the query templates. Our queries were focused on a few key pieces of disaster information. we gathered expert data about the specific subtasks each disaster had. For example, for earthquakes, we researched how to lift and remove rubble from a building collapse, and for the train derailment the annotators queried about the types of environmental testing that were done to detect dangerous chemicals in the area. We also queried about the specific objects used in each subtask, what they are used to achieve, and how to use them safely. Third, we researched precautions that should be taken for the disaster as a whole, both by civilians and by rescue workers. Without this expert knowledge, the templates would not be as useful or cover all relevant information. Examples of the resulting disaster-related questions that came from this research are in step 1 of Figure 1.

The annotators then inspected all of the brainstormed instructions, generalized over them, then wrote original instruction templates, as exemplified in step 2 of Figure 1. For the example “What can be used for travel and carry large loads,” the central notion (here, of having a task (travelling) that needs completion with the help of an object (a type of vehicle with the affordance of *go.02* INSTRUMENT) that has additional constraints that go beyond the basic affordance label (ability to carry large loads) was then “templatized” into prompts of the form, *Tell me which of these can perform [AFFORDANCE] given*

Category	Templates	Examples	Instances in Seed Sets
Relative Size/weight	Biggest Object, Heaviest Object, Relative Fit	Which of these objects is the lightest? outlet, broom, pail, orange, screen Would a shoe fit in a bag?	15
Appropriate Object Affordance	Basic Affordance, Size Restricted, Shape Restricted, General Property Restricted, Goal Restricted, Difference within Affordance, Difference within Affordance given Criteria	Which of the following can be used to climb and is bigger than a table? stile, stairway, stepladder, step, ladder What should I use if I want to learn something from the internet? What is the difference between a window and a pane?	38
Is-A and Hypernyms	Basic Is-A, Identical Usage, Sub-Types	Can you use a shed as a barn? List several types of truck and their use cases.	16
Objects in Risky Situations	Cause Injury, Cause Danger, Cause Object Damage	Which of the following objects would be the most dangerous if it hit something? dvd, screen, wall, drum, mat	16
Required Equipment	How to Use, Equipment for Scenarios, Role of Equipment in Task	Give a step by step explanation of how to use a concrete saw. What role does an air canister play in testing air quality?	15
Primary and Secondary Object Facts	Where Object Found, Objects in Location, Secondary Uses, Frequency of use, Average Knowledge of Use, Ease of Interaction Given Object State	Hey, which of the following can be used as a lever? art, motorcycle, picture, dvd, broom How well does the average person know how to use a concrete saw? Is a raised or lowered drawbridge more effective at getting cars across the river?	34
Disaster Specific Knowledge	Preparations, Warning Signs, General Information	List and explain the different hazards to look out for besides train cars after a train derailment.	10
Instruction Following	Instruction Identification, Follow-Up Questions	Choose the navigation instruction: drink from the bottle, sail a boat, enter the doorway	30

Table 1: An overview of the types of templates within each category, some examples of resulting seed sentences within each category, and the number of instances of each category within the resulting seed dataset. Note the emphasis on affordances, object knowledge, and instruction knowledge.

[GENERAL OBJECT PROPERTY]?. We then categorized this resulting template under the general category of “Appropriate object affordances” alongside other template instructions focused on querying about objects’ functionalities and affordances (see step 3 of Figure 1). The complete list of template categories with corresponding examples can be found in Table 1.

After developing the templates, the annotators used a list of objects from the disaster-specific expertise and labelled each object with all applicable PropBank sense-role pairings. We added these labels to Affordance Ontology previously described in 2.1. For instance, “Train,” which is relevant to the Ohio train derailment, was labelled *occupy.01* ARG1, *go.02* ARG2, and *contain.01* ARG0 by our annotators. This means a train can hold people, be used for transporting people, and can contain objects. “Air horn,” which is relevant to Earthquake search and rescue, was labelled with *signal.02* ARG0 and *alert.01* ARG1, meaning that an air horn can both signal information and warn of potential danger. This extension of the Affordance Ontology can be seen in step 4 of Figure 1. Examples of how Affordance Ontology labels connect to vocabulary used in the templates are in Table 2.

3.2. From Templates to Seed Instructions

For our next step, we determined what vocabulary could potentially fill in the blanks for each template. We examined each template and determined which vocabulary terms with associated linguistic properties from the list could appropriately fill in the blanks of each instruction. For instance, we determined that the affordance of *occupy.01* ARG1 (i.e. an object that a human can occupy) can appropriately fill in the AFFORDANCE slot for the template *Tell me which of these objects can perform [AFFORDANCE] given [GENERAL OBJECT PROPERTY]*. We then chose properties corresponding to each chosen sense-role label to fill in the GENERAL OBJECT PROPERTY slot, thus further restricting the number of correct answer objects. This process is shown in step 5 of Figure 1, where one exemplified PROPERTY slot associated with *occupy.01* ARG1 is *can move*, which restricts the list of potential correct answers from *balcony, barn, boat, building, car, floor (story), house, truck, train* to be *boat, car, truck, train*. Another exemplified property slot associated with *go.02* INSTRUMENT is *holds one person*, which restricts the resulting correct answers with the *go.02* INSTRUMENT affordance to only *motorcycle, bike*. This process of choosing appropriate affordances and properties for the Identical Use Case template is shown in Table 2.

We chose all possible vocabulary terms with associated linguistic properties for each template, then randomly selected which vocabulary items would fill in a particular blank to generate the final seed questions. An example of a final seed instruction, arising from the template "Tell me which of these objects can perform [AFFORDANCE] given [GENERAL USE CASE]" is "Tell me which of the following are places people can occupy and can move: car, building, train.". The resulting gold-standard instruction is seen in step 6 of Figure 1.

The linguistic annotators each decided on the correct answers based on context. For disaster related knowledge and required equipment knowledge, the annotators relied heavily on our disaster expert sources. In general, answers could not be automatically generated from templates because we often tested for linguistic knowledge that went more in-depth than the knowledge encoded in PropBank sense-role affordance labels. One example is in step 7 of Figure 1. Objects that have the label *occupy.01* ARG1 cannot be differentiated by mobility by affordance label alone. Similarly, in Table 2, sharing an affordance of *store.01* ARG2 does not indicate or preclude that "barn" and "shed" have a hypernym or is-a relationship. The annotators had to use their own common-sense capabilities to achieve the level of granularity we need for assessing LLM common sense capabilities.

		Populated by...
Template	Can you use [object-slot1] as a/n [object-slot2]?	Two objects w/ identical affordance
Potential Slot 1 Affordances	Path-of enter.01	**doorway**, **opening**, **gateway**, entrance, etc.
	ARG2 of store.01	**shed**, **barn**, **greenhouse**, silo, etc.
	Path-of go.02	**road**, **train track**, **floor**, doorway, trail, etc.
Potential Slot 2 Affordances	*same as above*	*same as above*
Seed 1	Can you use a **doorway** as an **opening?**	
Answer 1	**Yes** because a doorway is a type of opening found in buildings.	
Seed 2	Can you use a **shed** as a **barn?**	
Answer 2	**No** because a shed is too small to store hay, livestock, and tractors like a barn can.	

Table 2: Population of templates leveraging semantic role labeling linguistic features for quick generation of domain-specific seed sentences: The template requires two objects within affordances that annotators identified contain terms with hypernym relationships. Two objects with the same sense-role label, or affordance, are then randomly selected to fill each slot, and a linguistic annotator uses common sense knowledge to answer the resulting query. By training the model on both correct and incorrect answers that naturally arise from random generation, the deeper linguistic meaning of use-case hypernyms is expressed in our data.

Upon request, we will make available both complete sets of seed questions, which also serve as an evaluation set for the model tuned for an earthquake disaster and the Ohio train derailment, respectively. In Table 2, we demonstrate our workflow for developing the disaster-specific seed set efficiently for the Identical Use Case template. With our annotation workflow for developing new models for new disaster scenarios, we can use an expert's time to provide only disaster-specific questions and vocabulary, as well as rating existing template quality.

4. Resulting Datasets

Our resulting datasets balance between covering a wide variety of physical object properties, such as size and weight, and holding specific knowl-

INSTRUCTION: Choose the visibility related instruction.

OPTIONS: carry the suitcase, *look through the lens*, sit in the armchair

GPT ANSWER: carry the suitcase ✘

PROPBANK POWERED DATA ANSWER:

Look through the lens ✔

Figure 2: An example of output from our preliminary model developed using the earthquake PropBank Powered Data Creation dataset. Here, `text-davinci-003` (A version of GPT 3.5) fails to choose the correct instruction from the options, but our much smaller model with PropBank Powered Data Creation can successfully correlate visibility with the pertinent instruction.

edge for an LLM to draw from when generating synthetic data based on the dataset. Furthermore, the datasets thoroughly cover required information for two very different types of disasters. For earthquakes, the priority is rescuing trapped individuals and clearing away rubble and partially collapsed buildings. For the Ohio train derailment, the focus was on monitoring the air, water, and soil for dangerous chemicals and ensuring the volatile chemicals that leaked from the train cars did not explode.

We initially tested PropBank-Powered Data Creation with our earthquake seed sentence dataset. This was a lengthy process of determining the types of templates we wanted, what they would be, and what vocabulary fit with each template. In contrast, developing the seed sentences for the Ohio train derailment took about 10 hours because we built on the pre-existing templates and potential choices for each fill in the blank. We are now confident that a disaster expert would need to give an hour of their time and some pointers to relevant literature to make PropBank Powered Data Creation successful. An expert annotator would then need one work day to develop the seed sentences. This means that the time between interviewing the disaster expert and deploying a model using PropBank-powered data could be as little as 3-4 days, depending on computational fine-tuning resources.

5. Baseline Fine-Tuned Model

The next step in our research is to use the PropBank-powered data as in-context learning examples for generating a synthetic dataset that will, in turn, fine-tune a small language model. We have made a preliminary model using the PropBank-powered earthquake data as our seed sentences, `text-davinci-003` as the model that generated a synthetic dataset of 20,000 instructions (OpenAI, 2023), and the LLaMa 7B model for fine-tuning (Touvron et al., 2023). We then had evaluators with expertise in linguistics compare the outputs of `text-davinci-003` and our PropBank-powered model by voting for which LLM won or if there was a tie and rating the quality of the winning answer on a scale of 0-3.

While the model our team developed had some successes, as can be seen in Figure 2, our preliminary results show we still have work to do. We had 3 annotators vote in our head-to-head testing, which resulted in our model winning approximately 8% of the evaluation prompts, tying with `text-davinci-003` for approximately 22.5% of the prompts, and losing to `text-davinci-003` for approximately 69% of the prompts. Further investigation found this was likely due to poor alignment between the seed sentences and the synthetic data. We believe the poor alignment was due to insufficient in-context learning during the data generation process, and are looking to improve this in future iterations. Making a preliminary model did prove that PropBank Powered Data Creation can be used both as evaluation and as seeding data, and we are excited to explore those capabilities as well in future work.

6. Related Work

6.1. Evaluation Datasets for Robots

Ahn et al. (2022) tests LLMs' abilities to execute instructions by developing a set of tasks for the robot agent to learn using reinforcement learning, then training a model to calculate the probability of a task being completed successfully paired with the probability that a natural language instruction will precede a given task. To do this, the authors wrote 101 instructions addressing various degrees of semantic complexity, including following primitive instructions, abstract nouns and verbs, and long-horizon planning that requires many steps to accomplish the instruction. The model, called SayCan, developed skills that transfered from the mock kitchen where it was trained to a real kitchen with minor losses in planning and performance. More interestingly, the authors also showed that SayCan performed better when they used larger LLMs with more linguistic knowledge. They also were able to utilize chain-of-thought fine-tuning to get a natural language explanation about the tasks that SayCan executed in order to fulfill the instruction.

Rather than having the LLM create a policy for a robot agent to execute itself, Xie et al. (2023) have GPT 3.5 translate the premise of the instruction from natural language to Planning Domain Definition Language (PDDL), an explicit way of defining all objects, predicates, and available actions within an environment. To test GPT 3.5's abilities to translate tasks, the authors developed tasks related to block stacking and navigating a kitchen that test an LLM's basic parsing competence, object association between natural language and entities in PDDL, numerical reasoning, physical and spatial reasoning, and world knowledge. They found that GPT 3.5 was able to perform well when the instructions were completely explicit and had decent performance at filling in the blanks for specifying goals and had decent reasoning about basic real world objects and relations. However, the authors also found that GPT 3.5 could not handle the complex and ambiguous physical relationships, and that GPT 3.5 likely relied extensively on the one-shot example it as given, rather than reasoning about the domain as a whole (Xie et al., 2023).

6.2. Robots and Language Models

PaLM-E is a multi-modal model designed to accept image, text, and sensor data and then output images, answers, or plaintext robot policy (Driess et al., 2023). This is achieved by vectorizing images into the same space as text embeddings, which allows for multi-modal fine-tuning but makes it unclear how the model would determine a particular robot's capabilities. RT-2 takes PaLM-E a step further by encoding language, vision, and actions into the same embedding space (Brohan et al., 2023). This allows for the robot agent to go beyond making only policy to making specific moves.

Instruct2Act takes a different approach and trains a LLM to output python code for a closed loop of perception, planning and actions (Huang et al., 2023). It does this by supplying the LLM with a variety of APIs for completing perception and action tasks. The scope of testing was limited to table top simulations, but the framework is inherently more flexible because the model can be fine-tuned to produce different python code.

These models all elicit interactions with the physical world, but Ghaffari and Krishnaswamy (2023) argue that these connections can't fully capture the complexity of the physical world because they don't include any physical data beyond images. To solve this problem, they train a neural network on physical simulations, then create a LLM embedding affine transformation matrix from both the physical embedding space and GPT3 embeddings. They find that LLM embeddings in the physical embedding space do correlate with the objects they describe, Most interestingly, nouns have a stronger correlation, and are thus more grounded, than verbs and attributes, much like how nouns are often learned first during language acquisition (Ghaffari and Krishnaswamy, 2023).

7. Future Work

In addition to our immediate goal of improving synthetic data generation techniques and fine-tuning parameters, we are interested in expanding PropBank Powered Data Creation to become multimodal. While even smaller multi-modal models are still too large to be useful in our robotics domain, there is a clear path for the expansion of our protocol. Notably, we hope to gather image data that can reinforce what different objects may look like in a given environment, how to interact with relevant equipment, and objects performing their affordances or changing states. These images could be paired with PropBank labels, vocabulary terms, and complete instructions. The variety of ways images can be combined with PropBank Powered Data Creation makes this an exciting new avenue for improving transformer model performance on disaster scenarios.

8. Conclusion

We introduce PropBank Powered Data Creation, a pipeline for efficiently creating semantically motivated seed sentences to be used for generating synthetic data for disaster related scenarios. We extended our Affordance Ontology and created 2 sets of 175 seed sentences for the domains of earthquake search and rescue and chemical spills following train derailments. These seed sentences extensively query objects' affordances, physical characteristics, changes of state, and fine-grained properties to ensure thorough evaluation of a LLM trained on PropBank Powered Data Creation-based synthetic data. We created a LLM demonstrating this full pipeline, and will continue to work on aligning our synthetic data to our seed sentences to increase LLM performance in disaster-related domains.

9. Ethical Considerations

PropBank Powered Data Creation is fundamentally based on biasing a language model towards feedback from a small group of selected sources. While this is for a positive effect within our domain, it may be harmful in domains that require more social common sense than ours. Within our templates, we tried as much as possible to be gender-neutral to discourage gender bias.

Our biggest form of bias is in assumptions of the specifics of our objects. We imagined our objects from a Western perspective, which can affect the affordances assigned to the object and how we query the object's properties. For instance, we imagine "curtains" to be window dressings, but in nomadic cultures a curtain could be used to separate living spaces within a tent. A positive about the structure of PropBank Powered Data Creation is that it purposefully allows time for adding and editing to the Affordance Ontology in order to align the data to a particular disaster and location. However, this is time consuming and puts the onus on the linguistic annotator to adjust the ontology both quickly and with cultural sensitivity.

Though the domain of this project is robots in disaster relief scenarios, we have not tested any implementation of this dataset on a robot, let alone a robot in a dangerous situation. We caution that extensive grounded testing must be done on any LLM resulting from these data before any real-world implementation can occur safely.

10. Works Cited

Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Daniel Ho, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Eric Jang, Rosario Jauregui Ruano, Kyle Jeffrey, Sally Jesmonth, Nikhil J Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Kuang-Huei Lee, Sergey Levine, Yao Lu, Linda Luu, Carolina Parada, Peter Pastor, Jornell Quiambao, Kanishka Rao, Jarek Rettinghouse, Diego Reyes, Pierre Sermanet, Nicolas Sievers, Clayton Tan, Alexander Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Mengyuan Yan, and Andy Zeng. 2022. Do as i can, not as i say: Grounding language in robotic affordances.

Air Sampling. 2024. Air sampling data.

Adolfo Arranz, Simon Scarr, and Jitesh Chowdhury. 2023. Searching for life in the rubble: How search and rescue teams comb debris for survivors after devastating earthquakes.

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 *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*, pages 178–186.

Claire Bonial, Susan W Brown, Martha Palmer, and Ghazaleh Kazeminejad. 2021. The rich event ontology. *Computational Analysis of Storylines: Making Sense of Events*, page 47.

Claire Bonial, Lucia Donatelli, Mitchell Abrams, Stephanie Lukin, Stephen Tratz, Matthew Marge, Ron Artstein, David Traum, and Clare Voss. 2020. Dialogue-amr: abstract meaning representation for dialogue. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 684–695.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alexander Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil J. Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael S. Ryoo, Grecia Salazar, Pannag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong T. Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. 2023. RT-2: vision-language-action models transfer web knowledge to robotic control. *CoRR*, abs/2307.15818.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Derailment Tools. 2022. Derailment response.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, and Zhifang Sui. 2023. A survey on in-context learning.

Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. 2023. Palm-e: An embodied multimodal language model. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 8469–8488. PMLR.

Sadaf Ghaffari and Nikhil Krishnaswamy. 2023. Grounding and distinguishing conceptual vocabulary through similarity learning in embodied simulations. In *The 15th International Conference on Computational Semantics*.

Siyuan Huang, Zhengkai Jiang, Hao Dong, Yu Qiao, Peng Gao, and Hongsheng Li. 2023. Instruct2act: Mapping multi-modality instructions to robotic actions with large language model.

Hydraulic Rescue Tools. 2022. Hydraulic rescue tools: What are your options?

Ghazaleh Kazeminejad, Claire Bonial, Susan Windisch Brown, and Martha Palmer. 2018. Automatically extracting qualia relations for the rich event ontology. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2644–2652.

Ian Niles and Adam Pease. 2003. Linking lexicons and ontologies: Mapping wordnet to the suggested upper merged ontology. In *Ike*, pages 412–416.

OpenAI. 2023. Openai gpt-3 api [text-davinci-003].

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1):71–106.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? 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 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Matt Scarbury. 2015. Rescue methods structural collapse step cutting concrete.

Mollie Shichman, Claire Bonial, Austin Blodgett, Taylor Hudson, Francis Ferraro, and Rachel Rudinger. 2023. Use defines possibilities: Reasoning about object function to interpret and execute robot instructions. In *Proceedings of the 15th International Conference on Computational Semantics*, pages 284–292, Nancy, France. Association for Computational Linguistics.

Soil Sampling. 2024. Soil and sediment sampling data.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca. Technical report, Stanford University.

Thermal Cameras. 2023. Tic peripheral: Search camera head (thermal).

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language model with self generated instructions.

Water Sampling. 2023. East palestine train derailment information.

Yaqi Xie, Chen Yu, Tongyao Zhu, Jinbin Bai, Ze Gong, and Harold Soh. 2023. Translating natural language to planning goals with large-language models.

Holly Yan, Christina Maxouris, and Nicki Brown. 2023. The ohio toxic train wreck was '100% preventable' – but there's no evidence the crew did anything wrong, investigators say.

Aspect Variability and the Annotation of Aspect in the IMAGACT Ontology of Action

Massimo Moneglia, Rossella Varvara
University of Florence, University of Fribourg
Via della Pergola 60 50121 Florence (Italy), Avenue de Beauregard 13 1700 Fribourg (Switzerland)
massimo.moneglia@unifi.it, rossella.varvara@unifr.ch

Abstract

This paper highlights some theoretical and quantitative issues related to the representation and annotation of aspectual meaning in the IMAGACT corpus-based multimodal ontology of action. Given the multimodal nature of this ontology, in which actions are represented through both prototypical visual scenes and linguistic captions, the annotation of aspect in this resource allows us to draw some important considerations about the relation between aspectual meaning and eventualities. The annotation procedure is reported and quantitative data show that, both in the English and Italian corpora, many verbs present aspectual variation, and many eventualities can be represented by locally equivalent verbs with different aspect. The reason why verb aspectual class may vary is investigated. Our analysis makes once more evident that verbs may vary their aspectual properties with respect not only to their argument structure but, more precisely, to the inner qualities of the eventualities they express. Crucially, when eventualities are expressed by equivalent verbs with different aspectual properties, the verbs focus on different parts of the structure of the eventuality.

Keywords: action ontology, aspect, semantic variability

1. Introduction

Since Verkuyl (1972), the importance of considering argument structure in the analysis of verbal aspectual information has been frequently pointed out. Accounts that attribute unique aspectual classes to verb lexemes fail to capture the complexity of this semantic property. Verbs may show a unique aspectual class or vary with respect to their valency, different interpretations, and the properties of the eventualities they can denote.

This paper deals with the representation and annotation of verbal aspectual properties in the IMAGACT ontology (Moneglia et al., 2014), a multilingual and multimodal ontology of actions derived from English and Italian spoken corpora (Moneglia, 2014). This annotation lets us reconsider the nature of aspectual properties by deriving the aspectual class of each action verb in relation to the different actions it can extend, giving a measure of the quantitative relevance of aspect variability in language usage.

In particular, it becomes possible for English and Italian verbs to observe: a) variation of the aspectual class of a verb across the various action types it can extend; b) variation of the aspectual class in the same action type by *locally equivalent verbs*, which is the peculiar information provided by IMAGACT (Moneglia et al., 2018).

The paper is structured as follows: Section 2 introduces the IMAGACT ontology, describing the methodology used to annotate aspect (2.1) and reporting quantitative data on aspectual variations in two languages considered in the ontology, English and Italian (2.2). Section 3 analyses the cases in which a single verb shows variation in its aspectual class, together with a theoretical explanation of these cases. Section 4 addresses the aspectual variation observed on eventualities, i.e., cases where an action concept is expressed through locally equivalent verbs with different aspectual properties. In 3 and 4, we will go through the linguistic and cognitive factors that give rise to the two kinds of aspect variability. We will only consider emblematic cases taken from the English verbal lexicon, leaving Italian and complex crosslinguistic variability problems to other occasions. In section 5, we draw some conclusions and summarize our findings. Table 1 in the Appendix will list the verbal entries in the IMAGACT lexicon that record both event and process readings and the proportion between the two categories across the set of eventualities they can extend.

2. The IMAGACT ontology and the annotation of aspect

IMAGACT is a multilingual ontology of action that visually represents the meaning of verbs referring to physical actions through scenes rather than through linguistic definitions. Each scene represents the prototype of an action type in the form of a video or 3D animation.

Action concepts were identified by annotating Italian and English spontaneous speech corpora using a complex induction procedure (Moneglia et al., 2012). Starting from the contexts of occurrence of verbs related to physical actions, the different activities each verb can extend to were highlighted. Each considered action verb's occurrence was examined (around 600 action verbs per language that are high frequency in oral contexts). Occurrences referring to physical actions were selected and expressed in a *standardized sentence*, in which the verb is linked to the minimum number of arguments necessary to represent the action. Once all occurrences of the verb were processed, the meaning of each became clear in its standardization. The semantic variation of a verb is thus inducted from corpora.

5th Workshop on Designing Meaning Representations (DMR 2024) @LREC-COLING-2024, pages 11–19
21 May, 2024.

Reconciling the action concepts identified in the two corpora into a single ontology, a set of 1,010 scenes was generated, each representing a prototype of action. This set, being derived from corpora representative of oral use, ideally constitutes the universe of relevant actions in the current socio-cultural context and how languages refer to them.[1] For each prototypical scene, the set of verbs referring to the same action concept, which are called *locally equivalent verbs* (Moneglia et al., 2018), are then mapped.

In summary, the ontology provides two main pieces of information:

a) the variation of action verbs, often general, across different actions.

b) the set of verbs referring to the same action concept, which are *locally equivalent*.

Figure 1 provides an example of the variation of the general verb *to push*.[2] As the figure shows, each prototype can also be identified by at least another verb (reported below the figure), which is equivalent in extension to the verb *to push* for that particular case.

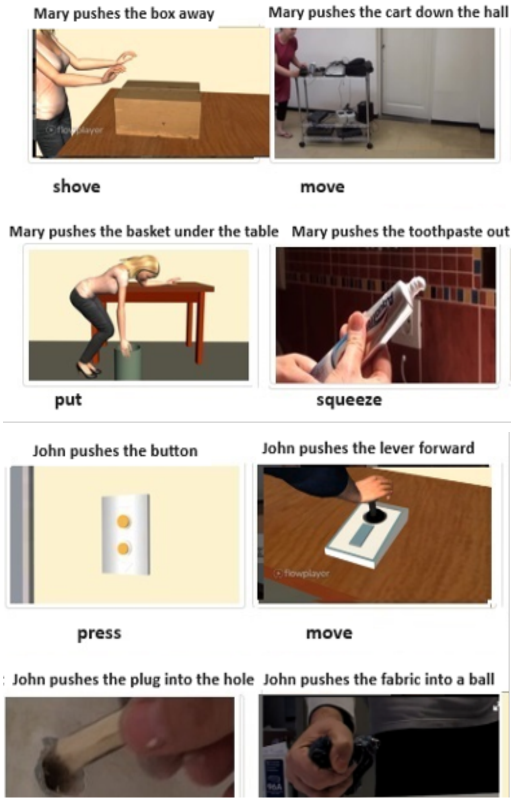

Mary pushes the box away
Mary pushes the cart down the hall
shove
move
Mary pushes the basket under the table
Mary pushes the toothpaste out
put
squeeze
John pushes the button
John pushes the lever forward
press
move
John pushes the plug into the hole
John pushes the fabric into a ball

Figure 1 The variation of push *across action types and locally equivalent verbs*

Each prototype scene is described by the best example, i.e., a linguistic caption (reported in Figure 1 above the frames). The best examples were annotated with the thematic structure[3] and the aspectual class that the verb determines in that linguistic context, respectively process or event according to the traditional Vendler's typology (Vendler 1967).[4] This procedure is described in more detail in the next subsection.

The sentences were then grouped into types based on two criteria:

a) Similarity to the best example chosen to represent the class (cognitive constraint)

b) Substitutability with verbal occurrences with the same locally equivalent verbs (linguistic constraint)

For example, standardized occurrences of the verb *push* are grouped into action types, each headed by a best example, as shown in the left box in Figure 4.

We refer the reader to Gagliardi (2014) for the quality assurances on the IMAGACT creation and annotation process.

2.1 The annotation of aspect

The *imperfective paradox* test (Bach 1986; Dowty 1977; 1979; Pustejosky 1991; Bennet-Partee 2004) was used to assign the aspectual class. The test identifies as processes all sentences formed with a certain verb conjugated in the progressive (PROG) that logically implies the corresponding sentence in the present perfect (PP). On the contrary, sentences formed with verbs that, conjugated in the progressive, do not imply the corresponding sentence in the present perfect are identified as events[5]:

- Processes: Prog (p) > PP(p)
- Events: Prog (p) >/ PP(p)

For example, the verb *push* identifies a process in (ex. 1) because it implies the corresponding present perfect, while the verb *climb* results in an event in (ex. 2) because the sentence does not imply the corresponding one in the present perfect:

[1] The following languages have been further implemented in IMAGACT through competence-based judgments (Brown et al. 2014): French, Spanish, Portuguese, Greek, German, Danish, Swedish, Chinese, Japanese, Hindi-Urdu, Arabic, Serbian, and Polish.

[2] The figures included in the paper constitute frames of the video or animation provided in IMAGACT. The complete scenes can be seen on the web interface of the ontology at the following link: http://www.imagact.it/imagact/query/dictionary.seam

[3] Relying on a modified version of the semantic roles inventory proposed by Palmer et al. (2005). See Moneglia & Varvara (2020) for details.

[4] Focusing specifically on actions, states are not considered in the ontology. Moreover, no granular distinction is made among the various aspectual classes falling within the categories of process and event. For instance, events encompass both achievements and accomplishments (Vendler 1967). Processes gather continual and iterative interpretations.

[5] "If x is V-ing entails x has V-ed, then either the verb or the predicate is a process."(Pustejovsky, 1991: 36)

1) Fabio is pushing the cart > Fabio has pushed the cart
2) Fabio is climbing onto the chair >/ Fabio has climbed onto the chair.

The test allows expert mother tongue annotators to easily attribute the aspectual class to the best examples of action types extended by all verbs in IMAGACT face to each action prototype, which ensures its actual interpretation.

This approach has generated a substantial database of correlations between the two aspectual classes and verbs. It becomes possible to obtain relevant data regarding the many verbs (not all) that exhibit aspectual variation in the different action types they can predicate.[6] For example, the verb *push* exhibits aspectual variation in Figure 2 between action type A (a process, as demonstrated by the paradoxical inference reported in ex. 3) and action type B (an event, as demonstrated by the lack of inference in 4):

3) Maria is pushing the cart > Maria has pushed the cart.
4) Maria is pushing the box >/ Maria has pushed the box.

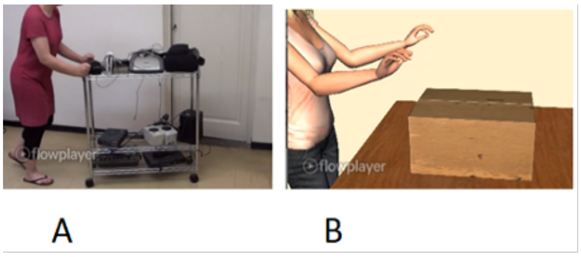


Figure 2 Two eventualities of the variation of to push.

However, in many cases, verbs with different aspectual qualities can identify the same action event. Considering the action in Figure 3 expressed by the verb *to push*, locally equivalent verbs can also be applied to that eventuality (*press*, *put*, *insert*), each one with different meanings and aspectual qualities, being either processive, like *push* and *press* in (5) or events, like *put*, *insert* and *place* in (6).

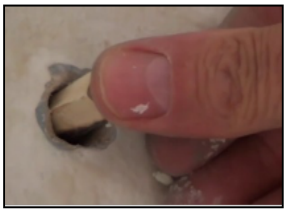


Figure 3 One of the eventualities ("John pushes the plug into the hole") expressed by to push, *locally equivalent to* press, put, insert.

5) Is pushing (pressing) the stick into the hole > has pushed (pressed) the stick into the hole.
6) Is putting (inserting) the stick into the hole >/ has put (inserted) the stick into the hole

Figure 4 illustrates how arguments are annotated, and the aspectual class is assigned to occurrences of Type 1 (in light blue on the left), where *push*, in the best example "*John pushed the stroller along the pavement,*" is locally equivalent to *move*, marking a process (in the central box). Similarly, in the annotation of Type 5, where *push* is equivalent to *shove*, the best example, "*Mary pushed the book away*" is marked as an event. The information concerning the possible aspectual class variation of a verb in the variety of actions is, therefore, a function of this level of annotation.

The actional concepts represented through visual prototypes must ensure that the ontological referring object for all locally equivalent verbs in that type is the same. For example, in the case of the verb *push*, the type corresponding to the *best example*, "*push the plug into the hole,*" must be mapped onto the same scene extended by the locally equivalent verb *insert*. This association provides information about actions that locally equivalent verbs can identify, getting, in some cases, different aspectual classes for the same scene.

2.2 Quantitative data

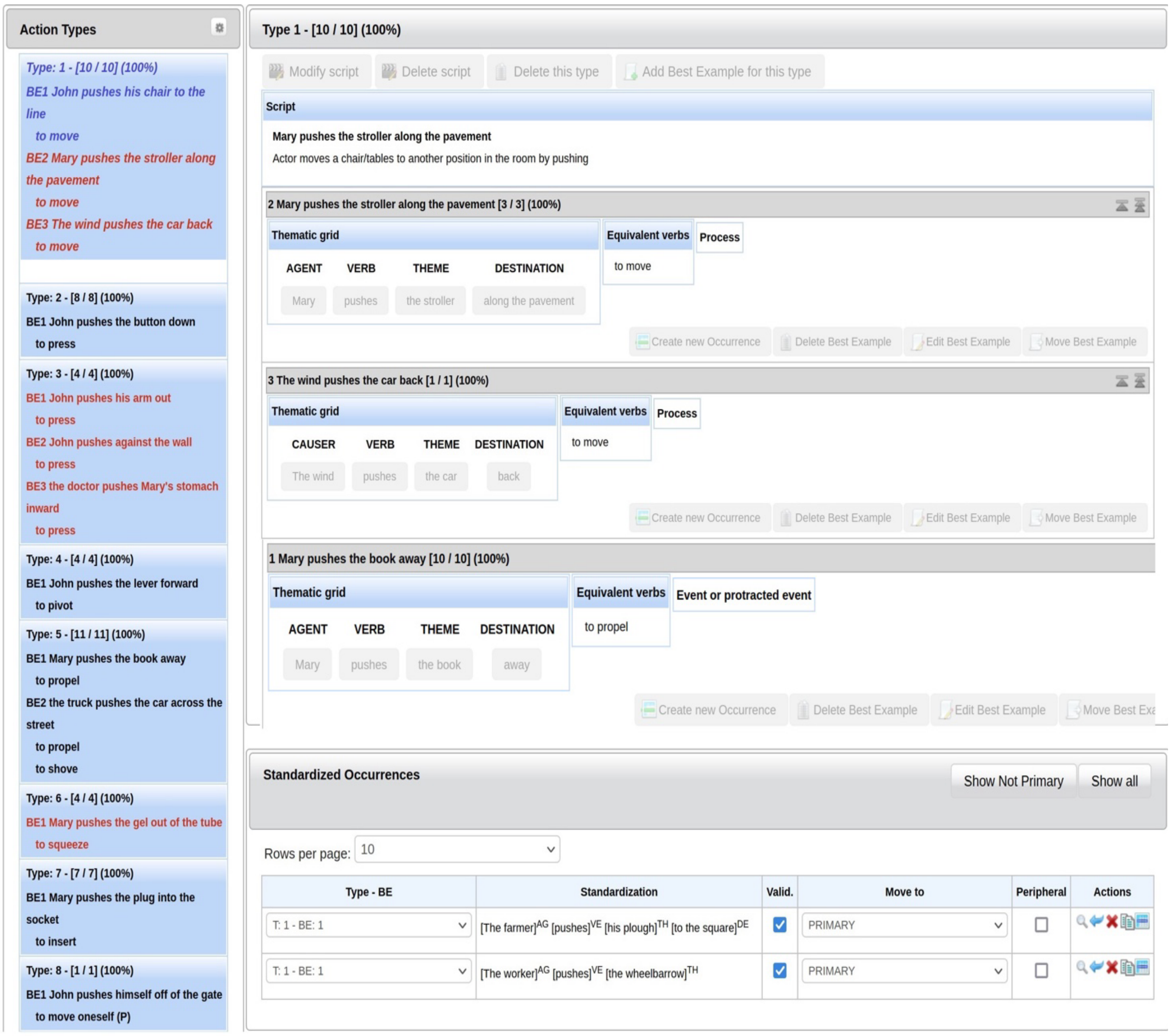
From this annotation, we can derive quantitative data from the IMAGACT database, which gives a measure of how aspectual variation impacts the interpretation of sentences referring to physical actions.

Considering the English lexical encoding, out of 543 verbs examined, 393 consistently remain in the same aspectual class (301 are always annotated as events and 92 as processes). In comparison, 150 verbs exhibit variation in the various types they are annotated with. Among the 943 actional types extended by these verbs, 640 are always identified by verbs conveying the same aspect: 478 are consistently extended by verbs annotated as events and 162 as processes. However, 303 action types can be identified by verbs with different aspects.

Similar results are observed when considering the annotation of Italian. Out of 501 annotated verbs, 401 never vary in aspectual class across the action types they extend to. Among these, 260 are marked as events and 141 as processes. The remaining 100 verbs exhibit aspectual variation in the different actions each can refer to. Considering the action types extended by the Italian verbs in question (920), 709 prototypes are identified by verbs that give rise to a single aspectual class (511 annotated as events and 197 as processes), while 211 action types can be extended by verbs that exhibit aspectual variation. The pie charts in Figures 5 and 6 illustrate the quantitative data.

In short, in English, one out of three action types in the ontology undergoes different aspectual categorization, and one out of four action verbs may change their aspect when applied to different action types. The slightly reduced proportions scored in

[6] English verbs that exhibit aspectual variation are reported in Table 1 in Appendix.



Action Types

Type: 1 - [10 / 10] (100%)
BE1 John pushes his chair to the line
to move
BE2 Mary pushes the stroller along the pavement
to move
BE3 The wind pushes the car back
to move

Type: 2 - [8 / 8] (100%)
BE1 John pushes the button down
to press

Type: 3 - [4 / 4] (100%)
BE1 John pushes his arm out
to press
BE2 John pushes against the wall
to press
BE3 the doctor pushes Mary's stomach inward
to press

Type: 4 - [4 / 4] (100%)
BE1 John pushes the lever forward
to pivot

Type: 5 - [11 / 11] (100%)
BE1 Mary pushes the book away
to propel
BE2 the truck pushes the car across the street
to propel
to shove

Type: 6 - [4 / 4] (100%)
BE1 Mary pushes the gel out of the tube
to squeeze

Type: 7 - [7 / 7] (100%)
BE1 Mary pushes the plug into the socket
to insert

Type: 8 - [1 / 1] (100%)
BE1 John pushes himself off of the gate
to move oneself (P)

Type 1 - [10 / 10] (100%)

Script

Mary pushes the stroller along the pavement
Actor moves a chair/tables to another position in the room by pushing

2 Mary pushes the stroller along the pavement [3 / 3] (100%)

Thematic grid				Equivalent verbs	Process
AGENT	VERB	THEME	DESTINATION	to move	
Mary	pushes	the stroller	along the pavement		

3 The wind pushes the car back [1 / 1] (100%)

Thematic grid				Equivalent verbs	Process
CAUSER	VERB	THEME	DESTINATION	to move	
The wind	pushes	the car	back		

1 Mary pushes the book away [10 / 10] (100%)

Thematic grid				Equivalent verbs	Event or protracted event
AGENT	VERB	THEME	DESTINATION	to propel	
Mary	pushes	the book	away		

Standardized Occurrences

Rows per page: 10

Type - BE	Standardization	Valid.	Move to	Peripheral	Actions
T: 1 - BE: 1	[The farmer]^{AG} [pushes]^{VE} [his plough]TH [to the square]^{DE}	☑	PRIMARY	☐	
T: 1 - BE: 1	[The worker]^{AG} [pushes]^{VE} [the wheelbarrow]TH	☑	PRIMARY	☐	

Figure 4 Interface for the Annotation of Thematic structure and Aspectual class of the best example of each Action Type

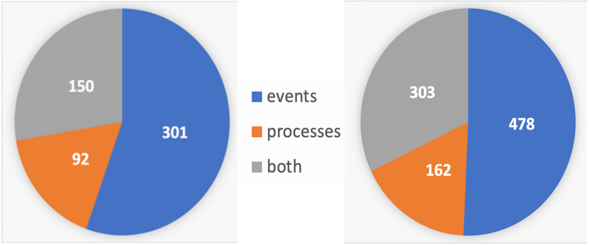


Figure 5 Aspect variability among English verbs (left) and types (right).

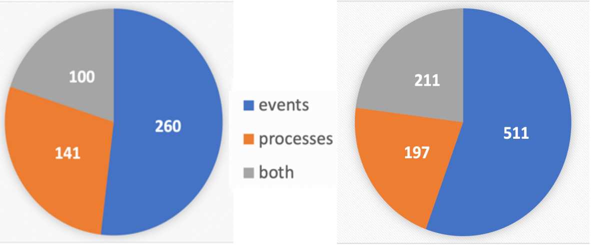


Figure 6 Aspect variability among Italian verbs (left) and types (right).

Italian do not change the overall picture.[7] Aspectual variation is, therefore, a quantitatively significant phenomenon when referring to actions. The definition of criteria by which a verb can give rise to an event or a process, or the same action can be seen as both a process and an event, is necessary to ensure natural language interpretation. In the following paragraphs, we will consider the factors influencing aspect variability.

[7] The reason for this variation raises complex questions concerning the cross-linguistic categorization of action concepts, but is not an object for this paper.

3. Aspect Variation of Verbs

3.1 Aspect Variation and Thematic Structure

In some well-known cases, thematic structure changes correlate with aspect changes. For instance, activity verbs (Dowty 1979), in their absolute structure, get an event interpretation when taking a thematic argument. For example, *to paint* in (7) and (8) respectively correspond to a process and to an event in prototypes A and B of Figure 7:

7) Mario paints > PROC[8]
8) Mario paints the hood TH > EVENT

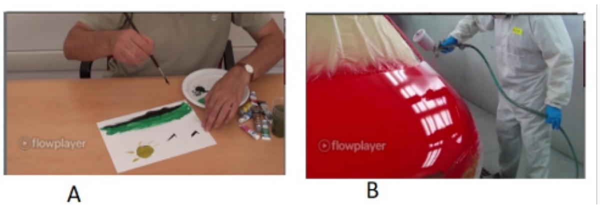


Figure 7: Aspectual variation of the activity verb to paint *(absolute vs non-absolute reading)*

Similarly, motion verbs, which are processes in their absolute structure, can exhibit aspectual variation when selecting an internal argument. For instance, the verb *to climb*, if it selects an internal argument with the role PATH (9), required when applied to prototype A of Figure 8, determines the processual interpretation. In contrast, the semantic role DESTINATION, required by prototype B, determines the event interpretation (10).

9) Fabio climbs the stairs PATH > PROC
10) Fabio climbs onto the chair DES > EVENT

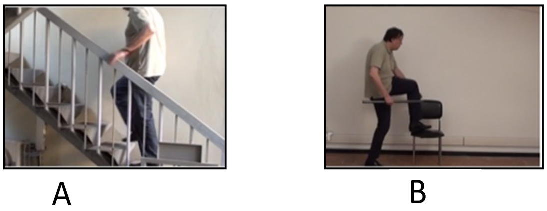


Figure 8: Aspect variation among two eventualities in the variation of to climb.

These cases are, therefore, predictable based on the minimal argument structure of the verb necessary for the projection of a specific action.

3.2 Aspect Variation of General Verbs across action types

IMAGACT demonstrates that the aspectual variation of a verb is not determined solely in relation to its argument / thematic structure but can also be due to the verb variation across action typologies. We have observed significant changes in the aspectual class of the clause in two paradigmatic cases:

a) Variations in the typology of the action extended by the same verb

b) Variations due to the pragmatic relevance of the resulting state

The first case is well identified in IMAGACT by those motion verbs that, in their proper meaning, can extend to both motion eventualities and eventualities in which the verb predicates of object relations.

Examples (11) and (12), depicted in Figure 9, illustrate the change in thematic structure (REFERENCE vs LOCATION) recorded by the verb *to pass*. The change occurs specifically when the verb predicates about a *motion in space* or, on the contrary, about *object relations*. In the first case, the truth of "*the guy is passing the light"* does not imply that he passed through, and the verb is an event in that eventuality. In the second case, the inference "*Mario passed the paint on the shelf*" holds, and nothing ensures the work is over.

11) Mario passes the light REF > EVENT
12) Mario passes the paint TH on the shelf LOC > PROC

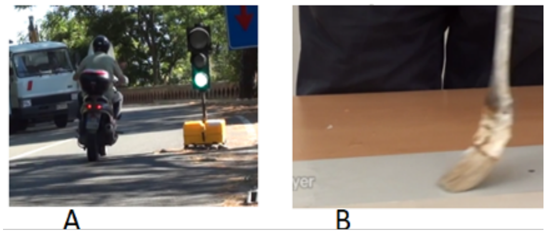


Figure 9: Aspect variation of to pass *in two eventualities*

Action verbs can undergo aspectual class variation depending on the greater or lesser relevance of the modification of the world achieved by the action. Consider, for example, the verb *tightens*. The sentences in (13) and (14), represented in the two prototypes of Figure 10, show that if the activity does not determine a relevant change of state, as in model A, the predicate has a processual interpretation, while it is interpreted as an event as soon as the activity is aimed at achieving functionally relevant goals, as in model B.

13) Fabio tightens the bottle > PROC
14) Fabio tightens the rope around Maria's neck > EVENT

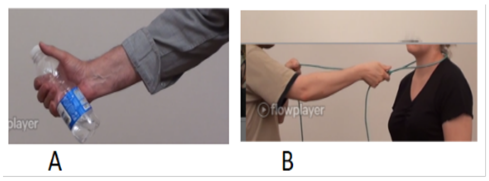


Figure 10: Two eventualities of the verb to tighten.

[8] For brevity, we leave it to the reader to replicate the assignment to the aspectual class through the test of the imperfective paradox.

Semantic correlations justify this variation. *Tighten* is a predicate that, when referring to scalar variations, has a processual interpretation as in (13), as pressure is exerted more or less without determining a final result. In fact, when "*Mario is tightening the bottle*", this implies that he has already tightened it a little bit.

However, the same verb takes an event reading when referring to events where a result emerges prominently, as in (14). "*Mario is tightening the rope around Maria's neck*" does not imply that he has tightened the rope around Maria's neck, which is true only in a state where the rope can be said to be tight.

We can replicate the phenomenon with other action verbs with a scalar application. For example, *to raise* can have a scalar reading or can apply to events in which the achievement of a relevant resulting state is predicated.

If I am raising the microphone, it can be inferred that it is already more or less raised (as in A of Figure 11), and the verb is a process. This is not the case in B, where it cannot be said that *Maria has raised the paddle* until the paddle is visible over her head, that is, until the state of functional relevance of the movement is reached, and the sentence refers to an event.

15) Maria raises the microphone > PROC
16) Maria raises the paddle> EVENT

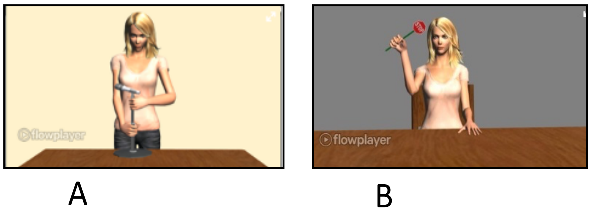


Figure 11: Aspect variation among two eventualities in the variation of to raise.

4. Aspect variation of equivalent verbs in the same eventuality

When considering that the same verb can vary its aspectual class in different eventualities, it seems straightforward the conclusion that aspect depends on the nature of the eventuality, which should be an entity within the natural language metaphysics, with the inner properties of a process or an event (Bach 1986). However, this conclusion cannot explain why the same eventuality can be interpreted as an event or a process when referred to by two locally equivalent verbs.

The phenomenon is relevant since, in English, it concerns one out of three of the eventualities represented in the ontology, as we observed above. For instance, consider the local equivalence between *to compress* and *to mash* (17 and 18, represented in the eventuality A in Figure 12) and between *to pour* and *to put* (19 and 20, represented in the eventuality B in Figure 12). *Compress* and *pour* lead to processive interpretations, while *mash* and *put* give rise to event interpretations of the same eventuality.

17) Fabio compresses the bottle > PROC;
18) Fabio mashes the bottle > EVENT
19) Maria pours the wine into the glass > PROC
20) Maria puts the wine into the glass > EVENT

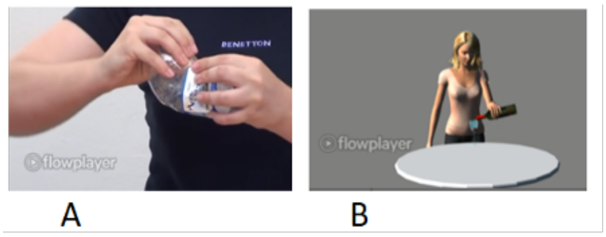


Figure 12: Two eventualities with equivalent verbs with different aspects

Given that the eventuality is one and only one, the explanation of this phenomenon can only be a function of the conditions of application of the verbs in question, i.e., the different semantics of these verbs. Therefore, we must consider both the semantic properties expressed by the verbs and how these relate to the properties characterizing the eventuality.

The structure of an event can be encoded as a transition between two states (von Wright 1963). In short, the event is a logical entity with two *foci*: '¬pTp', where T is the temporal transition of the state (*and then*) that produces the result (*the truth of p*) from a state in which *p* is not true. Reasoning in a pragmatic form, we can say that, in the domain of natural action, when ¬p is true, a set of acts (more or less prolonged in the sense of Vendler, 1967) occurs that lead to the result. ¬p and p are nothing but "entities of different kinds" in the sense of Bach 1986.

Considering the properties signified by the predicates, we identify the event's structure with the notation 'informative focus1 T informative focus2', to indicate that where ¬p is true, a set of positive pragmatic acts occur. The properties characterizing the semantics of the verb can, in principle, refer to the focus in 1, the focus in 2, or both foci of the event structure.

In other words, the existence of a positive focus on the resulting state, necessary to be an event, can not only be determined by what happens, as the impulse in Figure 2B or the pragmatic relevance depicted in Figure 9B. The emergence of an event reading can also depend on how a verb predicates an eventuality.

Considering the different semantics of the verbs applied to the eventuality in B of Figure 11, we can hypothesize that *put* is inherently resultative, as its information focus, i.e., the quality characterizing its meaning, is 'inserting an entity into a background' (Moneglia 2005). In other words, the meaning of the verb emphasizes the information focus 2 of the event structure, while it does not specify information about how this result is achieved (part 1 of the event structure). On the contrary, *pour* has an informational focus on the qualities of the object (liquids or mass entities) and the manner of the activity (*controlled*). Therefore, *pour* has an information focus in the first part of the event structure.

The same happens in the pair *compress/mash* in the prototype of Figure 12 A. As the variation of *mash* derived from IMAGACT in Figure 13 shows, this verb does not specify any information in the first focus of the eventuality. Indeed, forces that produce the result can be of whatever kind. *Mash* focuses on the information characterizing the result achieved, leading to the event interpretation of the eventuality.

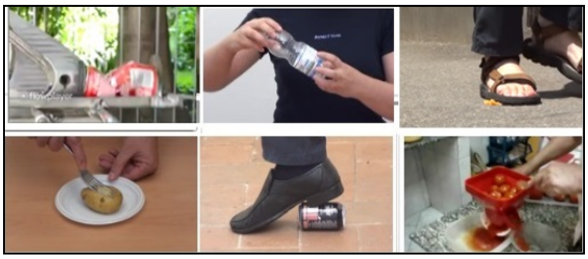


Figure 13: the variation of to mash *across action types.*

On the contrary, *compress* would indicate that the qualities of the forces exerted on the object are 'aimed at its reduction'. The object can result in being more or less compressed without necessarily reaching a final 'compressed' state.

This is clear from the comparison *of compress vs. mash* given by IMAGACT. *Compress*, but not *mash,* can be applied to elastic objects that cannot reach a permanently deformed state, as can be seen from Figure 13, where the actions denoted by *compress* and *mash* are compared (the first column comprehends actions denoted only by *compress*, the third column actions denoted only by *mash*, and the column in the middle shows actions that both verbs can denote).

Therefore, the verb meaning characterizes the information focus 1 of the event structure, resulting in a process interpretation.

5. Conclusions

The annotation of Aspect in IMAGACT is achieved in connection to the referential variability of action verbs, which can be synthesized as: "one verb many actions / one action many verbs". The resulting database sheds light on aspect phenomena, showing that aspect variability is a quantitatively relevant phenomenon impacting the interpretation of a good number of sentences referring to physical activities.

Variability regards the aspect of the same verb across different action types and the same action when referred to by different verbs.

The first phenomenon depends on the inner qualities of the various eventualities in the extension of one verb. When the relevance of a change of state emerges in activities showing continuity, such as movement and scalar forces, a granular distinction among action types is required, and the corresponding activity verb gets the event interpretation accordingly.

The second phenomenon involves lexical semantics. The different aspects conveyed by two locally equivalent verbs in the same eventuality tell us that their meaning picks up different properties of the same ontological entity. A verb can identify an eventuality indicating what happens in the process that leads to a result (information focus in the first part of the event structure) or, vice versa, the properties characterizing the result (information focus in the second part of the event structure). These are different ways to refer to an object (Frege 1892).

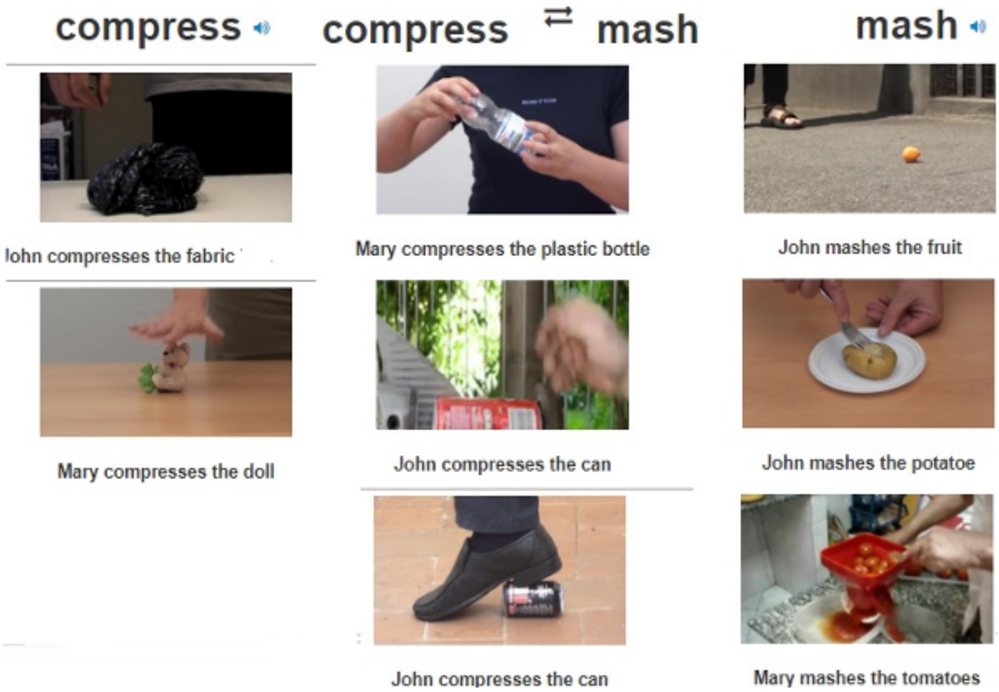


Figure 14: Comparison of the variation of to mash *and* to compress.

6. Bibliographical References

Bach, Emmon (1986). The Algebra of Events. *Linguistics and Philosophy* 9: 5-16.

Bennett, Michael / Partee H. Barbara (2004). Toward the Logic of Tense and Aspect in English. In: Partee H., Barbara (ed). *Compositionality in Formal Semantics*. Hoboken N.J.: Blackwell: 59-109

Brown, Susan / Gagliardi, Gloria / Moneglia, Massimo (2014). IMAGACT4ALL: Mapping Spanish Varieties onto a Corpus-Based Ontology of Action. *CHIMERA* 1: 91-135.

Dowty, David (1977). Toward a semantic analysis of verb aspect and the English “imperfective” progressive. *Linguistics and Philosophy* 1: 45–77.

Dowty, David (1979). *Word Meaning and Montague Grammar. The Semantics of Verbs and Times in Generative Semantics and in Montague's PTQ*. Dordrecht: D.Reidel.

Frege, Gottlob (1892). Über Sinn und Bedeutung, in *Zeitschrift für Philosophie Und Philosophische Kritik* 100 (1): 25-50.

Gagliardi, Gloria (2014). Validazione dell’ontologia dell’azione IMAGACT per lo studio e la diagnosi del Mild Cognitive Impairment (MCI). PhD Thesis, University of Florence.

Moneglia, Massimo (2005). Mettere. La semantica empirica del verbo di azione più frequente nel lessico verbale italiano. In: Biffi, Marco / Calabrese, Omar / Salibra, Luciana. *Italia Linguistica : discorsi di scritto e di parlato (Studi in Onore di Giovanni Nencion*i). Siena: Protagon Editori: 251-272.

Moneglia, Massimo (2014). The semantic variation of action verbs in multilingual spontaneous speech corpora. In Raso, Tommaso / Mello, Heliana (eds), *Spoken Corpora and Linguistics Studies*, Amsterdam: Benjamins: 152-190.

Moneglia, Massimo / Gagliardi, Gloria / Panunzi, Alessandro / Frontini, Francesca / Russo, Irene / Monachini, Monica (2012). IMAGACT: Deriving an Action Ontology from Spoken Corpora. In: Bunt, Harry (ed.) *Eight Joint ACL - ISO Workshop on Interoperable Semantic Annotation* (ISA-8). Pisa, October 3-5, 2012: 42-47.

Moneglia, Massimo / Brown, Susan / Frontini, Francesca / Gagliardi, Gloria / Khan, Fahad/ Monachini, Monica / Panunzi, Alessandro (2014). The IMAGACT Visual Ontology. An Extendable Multilingual Infrastructure for the Representation of Lexical Encoding of Action. In: Nicoletta Calzolari et al .(eds.), *Proceedings of the Ninth International Conference on Language Resources and Evaluation European Language Resources Association*, Paris: ELDA: 3425-3432.

Moneglia, Massimo / Panunzi, Alessandro / Gregori, Lorenzo (2018). Action Identification and Local Equivalence of Action Verbs: the Annotation Framework of the IMAGACT Ontology. In: Pustejovsky, James / van der Sluis, Ielka. *Proceedings of the LREC 2018 Workshop “AREA – Annotation, Recognition and Evaluation of Actions”*. Paris: ELDA: 23-30.

Moneglia, Massimo / Varvara, Rossella (2020). The Annotation of Thematic Structure and Alternations Face to the Semantic Variation of Action Verbs. Current Trends in the IMAGACT Ontology. In: Bunt, Harry (ed) 16th Joint ACL - ISO Workshop on Interoperable Semantic Annotation (ISA-16), The European Language Resources Association (ELRA): 68-75

Palmer, Martha / Gildea, Daniel / Kingsbury, Paul (2005). The Proposition Bank: An Annotated Corpus of Semantic Roles, *Computational Linguistics,* 31 (1): 71-106.

Pustejovsky, James (1991). The Syntax of Event Structure. *Cognition*, 41(1-3):47–81.

Vendler, Zeno (1967). *Linguistics in Philosopy*, Cornell University Press, Ithaca, NY.

von Wrigt, George Henrik (1963). *Norm and Action*. NY: Routledge.

Verkuyl, Henk J., On the Compositional Nature of the Aspects, Dordrecht, Reidel, 1972

Appendix

Table 1 List of English verbs with aspectual variation, with proportion of aspectual classes among the action concepts in the IMAGACT ontology.

Verbs	Event	Process
swing	0.09	0.91
smooth	0.10	0.90
brush	0.14	0.86
play	0.17	0.83
raise	0.17	0.83
draw2	0.19	0.81
rub	0.19	0.81
march	0.20	0.80
scatter	0.20	0.80
shorten	0.20	0.80
warm	0.20	0.80
smoke	0.22	0.78
dangle	0.25	0.50
paddle	0.25	0.75
pin	0.25	0.25
shine	0.25	0.75
trail	0.25	0.75
travel	0.29	0.71
chase	0.33	0.67
compress	0.33	0.67
dance	0.33	0.67
eat	0.33	0.67
follow	0.33	0.67
gallop	0.33	0.67
guide	0.33	0.67
lap	0.33	0.67
lean	0.33	0.67
lengthen	0.33	0.67

obstruct	0.33	0.33
scream	0.33	0.67
sew	0.33	0.67
shout	0.33	0.67
sleep	0.33	0.67
spin	0.33	0.67
squirt	0.33	0.67
stand	0.33	0.67
stroll	0.33	0.67
support	0.33	0.67
tow	0.33	0.67
water	0.33	0.67
block	0.36	0.27
gather	0.36	0.64
drag	0.38	0.62
feed	0.38	0.62
rotate	0.38	0.62
bend	0.40	0.60
boil	0.40	0.60
extract	0.40	0.60
ride	0.40	0.60
collect	0.44	0.56
rip	0.45	0.55
rest	0.46	0.15
roll	0.47	0.53
accompany	0.50	0.50
bear-2	0.50	0.50
circle	0.50	0.50
climb	0.50	0.50
cough	0.50	0.50
draw	0.50	0.50
extend	0.50	0.50
fry	0.50	0.50
hang	0.50	0.14
iron	0.50	0.50
knit	0.50	0.50
lash	0.50	0.50
light	0.50	0.50
pick	0.50	0.50
pound	0.50	0.50
puff	0.50	0.50
read	0.50	0.50
row	0.50	0.50
salt	0.50	0.50
surround	0.50	0.17
swim	0.50	0.50
tip	0.50	0.50
track	0.50	0.50
trot	0.50	0.50
whistle	0.50	0.50
widen	0.50	0.50
wind up	0.50	0.50
wrestle	0.50	0.50
yell	0.50	0.50
push	0.51	0.49
join	0.58	0.00
paint	0.60	0.40
squeeze	0.60	0.40
strain	0.60	0.40
weave	0.60	0.40
wind	0.60	0.10
wipe	0.60	0.40
tear	0.64	0.36
move	0.66	0.34
walk	0.66	0.34
connect	0.67	0.00
cry	0.67	0.33
drive	0.67	0.33
enclose	0.67	0.00
filter	0.67	0.33
hammer	0.67	0.33
knock	0.67	0.33
load	0.67	0.33
press	0.67	0.33
rise	0.67	0.33
scrub	0.67	0.33
seal	0.67	0.00
sing	0.67	0.33
sit	0.67	0.33
stride	0.67	0.33
sweep	0.67	0.33
type	0.67	0.33
wash	0.67	0.33
bring	0.70	0.30
crack	0.70	0.30
lay	0.70	0.00
spread	0.70	0.30
carry	0.71	0.29
ring	0.73	0.27
pour	0.75	0.25
stick	0.75	0.00
suck	0.75	0.25
tap	0.75	0.25
transport	0.75	0.25
tumble	0.75	0.25
pull	0.77	0.23
dust	0.80	0.20
lead	0.80	0.20
link	0.80	0.00
reach	0.80	0.20
restrain	0.80	0.20
run	0.80	0.20
write	0.80	0.20
toss	0.81	0.19
lower	0.83	0.17
connect up	0.85	0.00
copy	0.86	0.14
leave	0.86	0.14
open	0.86	0.07
fly	0.87	0.13
fall	0.88	0.12
remove	0.88	0.12
squash	0.88	0.12
crush	0.89	0.11
kick	0.89	0.11
throw	0.90	0.10
break	0.92	0.00
turn	0.92	0.08
hit	0.93	0.07
lift	0.93	0.07
put	0.94	0.06
give	0.95	0.05

NoVRol: A semantic role lexicon of Norwegian verbs

Henrik Torgersen[1], **Erlend Ø. Ravnanger**[1], **Lars Hellan**[2], **Dag T. T. Haug**[1]
[1] University of Oslo, [2] The Norwegian University of Science and Technology
hatorger@uio.no, erlenora@student.iln.uio.no, lars.hellan@ntnu.no, daghaug@uio.no

Abstract

In this paper, we describe NoVRol, a semantic role lexicon of Norwegian verbs. We start from the NorVal valency lexicon, which describes the syntactic frames of 7.400 verbs. We then enrich each of these frames by annotating, based on the VerbNet annotation scheme, each argument of the verb with the semantic role that it gets. We also encode the syntactic roles of the arguments based on the UD annotation scheme. Our resource will faciliate future research on Norwegian verbs, and can at a future stage be expanded to a full VerbNet.

Keywords: VerbNet, argument structure, semantic roles

1. Introduction

Semantic Role Labeling (SRL) is the task of identifying *Who did what to whom?*, i.e. what roles each of the argument entities bear in the event described by a predicate. Traditionally used for semantic representations, precise search, and in questions-answering systems, SRL has found new applications in the neural age, e.g., for image captioning (Chen et al., 2021) and computer vision (Sadhu et al., 2021), where it serves to structure the computer's interpretation of video. At the same time, the mapping from syntactic structure to semantic roles has also attracted considerable interest in theoretical linguistics with important contributions such as Fillmore (1968) and Levin (1993).

However, for Norwegian – otherwise a relatively well-resourced language – there are no datasets available that can support such research, whether practically or theoretically oriented. In this paper, we report on NoVRol, a resource which links the syntactic and semantic patterns of ca. 7.400 Norwegian verbs. For the semantic role annotation, we draw on the annotation standard of the English VerbNet (Schuler, 2005), with some modifications. For the syntactic side, we use the valency lexicon developed by Hellan (2022, 2023). In addition, we map these syntactic patterns to Universal Dependencies (UD, de Marneffe et al. 2021), thereby adding an important, lexical semantic resource to UD. UD currently containts more than 200 treebanks in more than 100 languages and has become the de facto standard for syntactic annotation and parsing. It is therefore a natural starting point for multilingual semantic parsing and many recent efforts in this direction have drawn on UD (Reddy et al., 2017; Poelman et al., 2022; Findlay et al., 2023).

We believe NovRol will be an important resource for future work in Norwegian NLP and linguistics. Moreover, because we follow the VerbNet annotation standard, we can expand the resource to a full VerbNet in future research by adding other information found in VerbNets such as selectional restrictions and event structure/logical form

The structure of this paper is as follows: in Section 2 we discuss related work on VerbNets and on the Norwegian valency lexicon. In Section 3 we describe the annotation procedure. Section 4 then discusses how our work fit in the broader picture of lexical resources for UD. Section 5 provides statistics about the data set, and Section 6 concludes and offers perspectives for further research.

2. Related work

2.1. Other VerbNets

The first VerbNet was developed for English (Schuler, 2005). It contains for each verb the semantic roles, selectional restrictions, syntactic frames and a semantic representation, as well as links to other lexical resources such as WordNet, PropBank and FrameNet. Also, verbs in VerbNet are organized in classes based on their valency alternation patterns, originally following the classes from Levin (1993) and later extended with more classes. The English VerbNet is therefore a comprehensive resource for the exploration of English verbs and their valency patterns. It has for example been used for the study of caused motion constructions (Hwang and Palmer, 2015). It has also been used in applications for word sense disambiguation, figurative language detection and it forms the basis for the semantic roles used in the Discourse Representation Structures of the Groningen Meaning Bank (Abzianidze et al., 2017). The latter was a particularly important motivation for our work, which is part of a project on UD-based semantic parsing.

There have been several efforts to create VerbNets for other languages, the most complete ones probably being those for Arabic (Mousser, 2010) and French (Pradet et al., 2014). Both of these started from the information in the English VerbNet and transfered this to the target languages semi-automatically. That is, they build on the idea that verb classes can be reliably identified across languages (see Majewska and Korhonen (2023) for a recent survey of this kind of work). This allowed for the relatively quick creation of rich resources with information comparable to that available in the original English VerbNet.

Our own approach was different, both because the goals were more modest – the immediate goal being a standard for semantic roles of Norwegian verbs for use in semantic parsing – and because Norwegian already has a rich resource for verbal valency, NorVal. It was therefore more natural to start from this Norwegian-specific resource and add information about semantic roles based on the English VerbNet, even if this meant that we gave up on structuring the resource around valency classes as in the English VerbNet and also do not provide much of the other information such as semantic structure or selectional restrictions. Some of this information is available in NorVal and can be more properly integrated in this resource to yield a richer VerbNet. We will come back to these opportunities later.

2.2. NorVal

NorVal (Hellan, 2022, 2023)[1] is a resource representing valency properties of 7,400 Norwegian verbs, theoretically based on the formal model outlined in Hellan (2019), and developed in parallel to a computational grammar of Norwegian, *NorSource*,[2] from which the verb inventory and many of the formal specifications have been ported.

The resource identifies 340 types of valency frames covering the valency properties of the verbs, and identifies for each verb lexeme which valency frames it can take. A compact notation system called *Construction Labeling* (abbreviated 'CL'), is used for classifying the frame types. More than half of the verbs take more than one frame, and the construct ⟨Verb, Valency frame taken by the verb⟩ is called a 'lexically instantiated Frame Type', abbreviated *lexval*. In the overall system there are currently 17,200 lexvals distributed over the 7,400 lexemes. Each lexval is illustrated by a 'Minimal Sentence' instantiating the lexval. A set of lexvals belonging to the same lexeme is called a *valpod*. To illustrate these constructs and their notation, (1-b) is the CL representation of the construction type: 'Expletive subject – direct object - extraposed declarative clause', exemplified by the verb *ane* ('dawn on') in (1-a):

(1) a. Det aner dem at krisen
it.expl dawns them that crisis.def
kommer
comes
'they have a hunch that the crisis is coming'
b. *trExpnSu-expnDECL*

The part 'trExpnSu' of this label is called the 'global label' of the lexval, indicating the valency frame as a whole (viz., *transitive with an 'extraposed' clause linked to subject position*), and the part 'expnDECL' is called an 'argument label' as it specifies one of the arguments.

The full set of constructions in which *ane* can be used, i.e. its valpod, is shown in Table 1. A valpod is verb-specific, but if one abstracts away the lexical item, one gets what may be called a *valpod type*, characterized by the set of frame types; such sets may be compared across the lexemes, and may be expected to provide a step toward a modeling of the notion of *verb classes* in VerbNet, based on defining valpod types across verb lexemes where a high degree of overlap in the members constituting a given set of valpods will qualify the lexemes characterized by these valpods for membership in a verb class.

NorVal provides syntactic frames for verb lexemes. Homonyms are distinguished in the verb list by hyphenated numbers, so that, e.g., *koste-1* represents the lexeme with meaning 'cost' and *koste-2* represents the lexeme with meaning 'brush'. Sub-senses of lexemes, on the other hand, are not originally recognized, but with the role annotation of this project, many cases are represented through added lexvals. Many aspects of what may be called 'basic logical form' are reflected in the frame type labels, such as causativity, semantic government, and infinitival control, and, most relevant to semantic role labeling, *participant* status, with semantic role features for *directionality* and *locativity*.[3] For example, the construction in (2-a) has the CL formula in (2-b).

(2) a. katten smyger seg langs muren
cat.def slithers refl along wall.def
The cat slithers along the wall.
b. *tr-obRefl-obDir*

This illustrates how a role specification is made by

[1] https://github.com/Regdili-NTNU/NorSource/tree/master/NorVal_files

[2] Hellan and Bruland (2015), Beermann and Hellan (2004), https://regdili.hf.ntnu.no/linguisticAce/parse

[3] These features are illustrated in the parse interface for NorSource at https://regdili.hf.ntnu.no/linguisticAce/parse, using the 'Minimal Recursion Structure' ('MRS') (Copestake, 2002) formalism for semantic representation; for a sentence like *Gutten løper til huset 'the boy runs to the house'*, for instance, a feature 'role' indicates the variable representing *huset* as 'endpoint', reflecting the combination of specifications of the verb and the preposition.

lexvals	explanation
ane intr	intransitive
ane tr	transitive
ane tr-obDECL	declarative complement
ane tr-obINTERR	interrogative complement
ane tr-suDECL	declarative subject complement
ane tr-suINTERR	interrogative subject complement
ane trExpnSu-expnDECL	transitive with expletive subject and extraposed declarative complement
ane trExpnSu-expnINTERRwh	transitive with expletive subject and extraposed wh-interrogative complement

Table 1: valpod for *ane*

appending the role indicator (*Dir*) to the argument label (*ob*), indicating that the object plays a directional role. The system also defines labels like *suAg* (subject agent), *suTh* (subject theme) and *obTh* (object theme),[4] and therewith valpods such as (3).

(3) a. *<V intr-suTh, V tr-suAg-obTh, ...>*
b. *<V intr-suAg, V tr-suAg-obTh, ...>*

The constellation in (3-a) could be used to characterize transitivity alternations like those found with verbs like *break*, as in *he broke the glass* vs. *the glass broke*, and the one in (3-b) to characterize alternations residing in constructions of 'object implicitation' like in *he is eating* vs. *he eats the bread*. While NoVRol uses a different notation, it provides a full scale encoding of roles for most aspects of verb semantics. Thus, two-membered valpods alone obtain for 1,500 verbs in NorVal, and many of them could be characterized as either of the options in (3). An assembly of valpods so annotated would throw interesting light on how common either of these types of transitivity alternations are in a representative valency inventory of a language. This illustrates how semantic role annotation, as undertaken in this project, provides an interesting addition to the specification inventory of NorVal.

3. Annotation

NoVRol includes every lexval in the NorVal database. Each verb and its arguments, as indicated in its lexvals, was annotated semantically according to the annotation guidelines for the English VerbNet.[5] The valpod for *ane* from Table 1 is shown annotated in Table 2.

We see that sometimes a single lexval needs to be assigned multiple semantic frames. For example, *ane tr(ansitive)* can take both an experiencer subject and a stimulus object and the inverse mapping. This is a special case because there is no associated meaning difference; in many other cases, the verb meaning changes slightly. For example, the verb *fortelle*, just like English 'tell' has among its syntactic frames one where it takes a subject, an object and a complement clause, but semantically, these can be agent–recipient–topic ('He told us that…') or pivot–experiencer-topic ('This tells us that…). Such multiple semantic frames are a major source of interannotation disagreement, as we will see below.

This yields a database of verb classes according to semantic roles, but without the in-depth listing of syntactic configurations or event structure specification provided by the English VerbNet. These are both aspects that can be added at a future stage. For the purpose of VerbNet as a lexical resource for a syntactic parser, this strategy has the advantage of allowing for the quick annotation of a large number of verbs. A test set of 800 (ca. 5% of total) verbs was reserved for evaluating inter-annotator agreement. In addition to the role annotation, we also give the Universal Dependencies labels for the different arguments. This section outlines how the annotation was done and comments on certain aspects of the results: differences between English and Norwegian; semantically ambiguous slots; inter-annotator agreement and the advantages and drawbacks of the annotation strategy.

3.1. Guidelines for annotation

The annotation process is split in two parts: semantic role assignment and assignment of Universal Dependencies Relations. Semantic role assignment in NoVRol is based on the annotation guidelines for the English Verbnet. In addition to annotating the verbs based on the guidelines, Norwegian verbs were compared with English translations and the semantics of their assigned VN classes to verify semantic similarity. In cases of inter-annotator disagreement, English VN classes were consulted for semantic properties to disam-

[4] This system for semantic annotation is extensively used in a resource for the West African language Ga, described in (Hellan, 2023) along with situation type labels. An issue for the annotation in that project was that many labels that had been used in similar applications for English were not adequate for Ga. We have not encountered similar issues in the present context, but, as a reviewer points out, this is an essential concern to keep in mind when classification systems in this area are borrowed from one language to another.

[5] https://verbs.colorado.edu/verb-index/VerbNet_Guidelines.pdf

lexvals	roles	UD
ane intr	experiencer	`nsubj`
ane tr	experiencer–stimulus	`nsubj--obj`
ane tr	stimulus–experiencer	`nsubj--obj`
ane tr-obDECL	experiencer–stimulus	`nsubj--ccomp`
ane tr-obINTERR	experiencer–stimulus	`nsubj--ccomp`
ane tr-suDECL	stimulus–experiencer	`csubj--obj`
ane tr-suINTERR	stimulus–experiencer	`csubj--obj`
ane trExpnSu-expnDECL	formal–experiencer–stimulus	`expl--obj-csubj`
ane trExpnSu-expnINTERRwh	formal–experiencer–stimulus	`expl--obj-csubj`

Table 2: Valpod for *ane* annotated for semantic roles and UD frames

biguate semantic role assignment. For example, the annotation of *hånflire* 'smirk', was annotated respectively as <agent, patient> (following the English class `bully-59.5`) and <agent, stimulus> (following `nonverbal_expression-40.2`). Only the latter class allows an interpretation where the verb is a reaction to a stimulus, which aligns with the usage of *hånflire*. The annotation <agent, stimulus> was chosen.

3.2. Annotation differences between Norwegian and English

Certain aspects of the English VerbNet do not straightforwardly align with Norwegian, or contain certain inconsistencies that this project dealt with. This section discusses three such examples.

Reflexives The annotation in NorVal pertains to syntactic properties exclusively, and not the possible status of *seg* as a semantic argument, i.e., a role-bearer; thus, *seg* in *skamme seg* 'be ashamed' is counted as an object on syntactic grounds, but would by most linguists be regarded as semantically empty. These are annotated as *null-role* in NoVRol.

One standard criterion for deciding the status as role-bearing vs. empty is substitutivity, i.e., whether another expression could be used in the place of *seg*. For example, *seg* in *skamme seg* cannot be replaced by another NP.

Another, less clear-cut, criterion is whether the situation type expressed by the construction 'feels' as expressing a participant corresponding to the position of *seg*. For *skamme seg*, this criterion matches the criterion of substitutivity. In contrast, the situation expressed in *Jon vasker seg* 'Jon washes himself' might be perceived as having just a single participant performing some activity, and thus implying no extra role status corresponding to *seg*; however, the object position is here fully substitutable by other NPs. In such cases the annotator will follow his or her intuition as to whether to assign a role or not to the reflexive.

In the English VerbNet, where the presence of light reflexives is far less prominent than in Norwegian, the annotation of reflexives in some cases makes use of the predicative relation `equals`. This relation is used in some <agent, patient> verbs, for example *dress oneself* (`dress-41.1.1`), to indicate that multiple arguments have the same referent. The predicate is absent from <agent, benefactive> verbs, e.g., *cook oneself a meal* (`preparing-26.3`), where the role annotation is the same as in the NoVRol.

Different role names The English VerbNet includes the roles *causer*, *circumstance*, *eventuality* and *subeventuality*. These roles are used in the database, but not mentioned in the documentation. *causer* has been annotated as having the possibility of being both cause and agent. *circumstance* is annoted as source. *eventuality* is annotated as theme, and *subeventuality* as co-theme. Subject expletives are given a *formal* role whereas they are just ignored in the English VerbNet. As mentioned above, light reflexives are annotated as *null-role*. These dummy roles facilitate the matching to UD syntax.

Directionals The English VerbNet contains multiple syntactico-semantic frames for structures that include directionals, whose adjunct/argument status is not clear in the literature (see for example Needham and Toivonen 2011 for discussion). One example is *pour* where the frame `pour-9.5` gives the following example: 'Maria poured water from the bowl into the cup'. In this example, there are two directionals introduced by prepositions. *The bowl* is annotated as *initial location* and *the cup* as *destination*. In NoVRol, we annotate such directionals with a lower degree of precision than other arguments, namely by the role tag *orientation*. The reason for this is that the exact role of such PPs largely depend on the semantics of the preposition itself, rather than that of the verb. Similar considerations led the Groningen Meaning Bank (Bos, 2013) to annotate the semantic role on the preposition itself.

Also, most verbs that can take directionals can take destination, source and path specifications, or any combinations thereof, yielding six different frames. Because directional adverbials are often interchangeable and combinable, this annotation shortcut is a more efficient way to preserve the information. In an SRL system, this information could then be used in combination with a lexicon of preposition senses to derive the actual semantic role in context. Moreover, the NorVal lexvals *suDir*, *obDir* and *PresntDir* tell us whether the direction specified is that of the subject, the object or the logical subject in a presentation construction, enabling a detailed semantic representation of the event structure. The task will remain challenging, however, as there are many ambiguous cases. For example, the verb *hoie* 'scream/yell' contains the lexval *intr-suDir*, which maps to the semantic tag *orientation*, which is ambiguous between different directionals, which could be realized by the preposition *etter* 'after, (here) at'. However, *hoie* also has an entry as a phrasal verb with the preposition *etter* 'scream/yell for', in which case the object of the preposition is invariably understood as a *topic*. Therefore only contextual knowledge can disambiguate examples like (4).

(4) De hoiet etter en lege
they screamed after a doctor
'They screamed at/for a doctor'

However, when the verb does not have a non-directional frame with a preposition that can introduce a direction, the *orientation* role makes it possible to retrieve the semantics of directionals.

3.3. Inter-annotator agreement

To evaluate the annotation quality, we set aside a test set of 800 lexvals, roughly 5% of the total lexval database size. These verbs were annotated by both annotators without discussion between them. All instances where the semantic frames differed in at least one semantic role were counted as disagreement. This could happen if the two annotators had assigned a different role to one of the arguments, irrespective of the number of semantic roles they agreed on. Another frequent error source are ambiguous verbs where the annotators had annotated two different frames, which were eventually both regarded as correct. Our metric is therefore relatively harsh, and the inter-annotator agreement rate was 0.58 measured using Cohen's kappa, which is relatively low. We nevertheless think the annotation is of high quality, as a closer analysis of the annotation mismatches reveals.

Of the 339 annotation mismatches in the test set, 41% of all mismatches were associated with verbs with multiple senses. Annotators had assigned different semantic rolesets, but the assigned rolesets were all valid. The verb *senke*, for example, may mean both 'sink' (<agent, patient> following the English class `other_cos-45.4`) and 'lower' (<agent, theme> – `put_direction-9.4`). Similarly, the verb *overtrekke* may mean both 'withdraw too much' and 'coat', fitting both `funnel-9.3` and `spray-9.7`.

In the remaining cases, different verb sense interpretations could not account for annotation mismatches. 55% of the remaining mismatches were yet categorized within the same macro-roles outlined in the VerbNet guidelines[6]. For example, the complement of the verb *overutstyre* 'overequip' was annotated respectively as *destination* and *recipient*, both members of the macro role *place*.

We conclude that most of the errors involve either annotators missing out on frames that should be present, in which case they can be added later, or they disagree on the exact role but agree on the macro-role, which means that even the wrong annotation is not too far off.

3.4. Annotating UD syntax

General strategy In addition to the semantic annotation, the verbs in the dataset were annotated for syntactic relations based on the UD scheme. This annotation was done for the 340 distinct valency frames in NorVal. Whenever possible, the annotation in the Norwegian UD treebank was consulted. Although most verbs in NorVal are not represented in the treebank, it was possible to find at least one verb from a particular frame most of the time. In doing this, we only paid attention to the syntactic labels assigned to the (heads of the) arguments. So for example, both interrogative and declarative complement clauses get the label `ccomp` in UD, and therefore the two NorVal frames *trExpnSu-expnDECL* and *trExpnSu-expnINTERRwh* get mapped to the single UD frame `expl--obj-csubj`. Similarly, UD does not distinguish subject and object control infinitives, while these are distinguished in the NorVal frames. As a result, the 340 NorVal frames are reduced to 64 UD frames, which therefore contain less information. However, while this is a lossy many-to-one mapping, the NoVRol does contain information about what NorVal frame the UD frame came from, making it possible at a later stage to extract more information and enrich the UD frames.

The UD frames of verbs are ordered by a hierarchy loosely following the Norwegian word order, as in (5).[7]

(5) `subj` $\prec$ `iobj` $\prec$ `obj` $\prec$ `advmod` $\prec$ `obl` $\prec$ `xcomp/advcl` $\prec$ `ccomp`

`subj` is not a UD relation, but a cover term for `nsubj`, `csubj` and `expl`, which in Norwegian is generally subject expletives. One exception to the

[6] https://verbs.colorado.edu/verb-index/VerbNet_Guidelines.pdf, p. 18

[7] See below for why some apparent adjunct functions are included in the valency.

above hierarchy happens when expletives cooccur with a displaced subject, which is called a 'logical subject' in traditional Norwegian grammar and is labelled `c/nsubj` in UD, although it occurs in object position (6).

(6) Det vil tilflyte oss penger
expl will flow us money
'There will flow money to us'

Such cases get the UD frame `expl—obj—nsubj`. Finally, the syntactic annotation was aligned to the semantics by arranging syntactic functions and semantic roles in the same order so that the mapping from function to role is transparent.

Adjuncts and obligatory arguments It is a common pattern for infinitival clauses in Norwegian to be introduced by prepositions, as in (7).

(7) Han ba dem om å gå
he asked them about to go
'He asked them to go'

Such infinitival clauses are treated as adverbial clauses (`advcl`) in the Norwegian UD treebank. This label suggests that they do not belong to verb's valency frame at all, but are adjuncts. This is clearly not the case, however. A related problem arises with nominal arguments, since UD does not distinguish arguments and adjuncts, but lump non-core (not subject or object) dependents as `obl`iques. These will be given a semantic role in our annotation if and only if they are considered arguments in NorVal and appear in the frames there. This means that when our lexicon is used in conjunction with a UD parse, one cannot know a priori whether an `advcl` or `obl` dependent will be assigned a semantic role or not. We see no way around this problem as long as UD does not distinguish arguments and adjuncts, since it is not practicable to list adjunct roles in a verb-based lexicon.

4. Lexical resources for UD

The UD initiative – and dependency treebanks in general – have historically been connected with the success of data-driven dependency parsing, which by its very nature required the annotation of running text rather than lexical resources. Dependents are annotated ”as they occur” and there is no attempt to extract more systematic patterns, unlike grammar-based parsers based on Head-Driven Phrase Structure Grammar (HPSG), Lexical-Functional Grammar (LFG) and Combinatory Categorial Grammar (CCG), which are typically based on rich lexicons. This move vastly improved robustness, but currently the very success of dependency parsing is sparking new interest in dependency grammar as a theory, which from its origins in Tesniere (1959) was always interested phenomena such as valency. We believe the time has therefore come to enrich UD with lexical resources.

Some moves in this direction are already seen within UD itself. For example, the UD validator relies on a list of auxiliary verbs which are actually annotated with a simple semantics, where they are marked as either Copula, Perfect, Past, Future, Passive, Conditional, Necessitative, Potential, Desiderative, Other or Undocumented auxiliaries. High-level information like this may be all that is possible to achieve at a universal level, although one can hope that it can be extended to other functional categories such as determiners, negators and subordinators.

More realistically, though, the creation of lexical resources will happen at a language-specific level and link up to the UD scheme. This is how we see the present contribution. However, rather than extracting information from a UD treebank and systematize and curate it to produce a lexicon, we have taken the information from resources built around the Norwegian HPSG grammar, which has been developed over two decades. Such resources, which have been handcrafted for many languages, but are often tied to specific linguistic formalisms (often LFG, HPSG or CCG) and even specific computational implementations of those formalism, contain a wealth of information that can be useful also in a dependency grammar context if it is made accessible in more theory-neutral forms as free-standing resources, alongside their function inside more closed systems such as computational grammmars. In particular, such handcrafted lexical resources contain a lot of information about the long tail of rare items: as stated above, NorVal contains ca. 7,400 verbs. By comparison, the first 10M tokens of the NoWaC corpus[8] contains 5,465 distinct verbs, the first 100M contains 6,929, and only the full corpus of 687M tokens surpasses NorVal and has 7,706 verbs.

5. Dataset statistics

The annotated verb set yields a database where syntactic features are given semantic tags. This section outlines the characteristics of the verb classes, their size, content and relations to syntax.

5.1. Number of classes

In our annotation, each lexval has been associated with a set of semantic roles, one role for each of the semantic arguments expressed in the frame. Such a set we may refer to as a *roleSet*; for each lexval, we may refer to its roleSet as a *lexvalRoleset*, and a roleSet abstracted away from its lexvalRoleset may be called a *roleSetType*. Across all the annotated lexvals, 250 roleSetTypes are used, and

[8] Norwegian Web as Corpus, Guevara (2010)

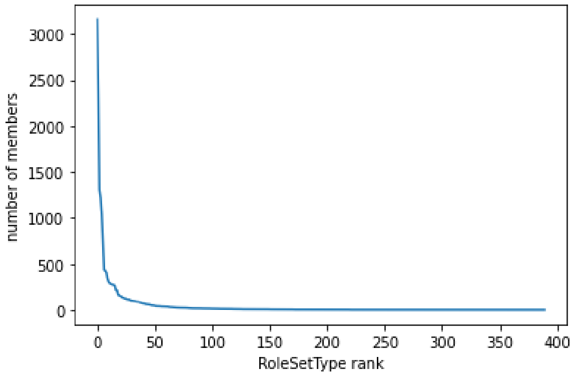


Figure 1: RoleSetType rank by members

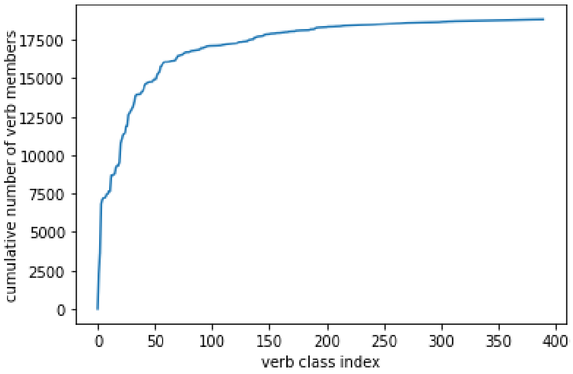


Figure 2: Cumulative members of by verb class index

we may define the notion *classes of lexvals* according to which roleSetTypes are aligned with the lexvals. Semantic role order is preserved – verbs annotated for the same semantic roles in different order, e.g., *fear* and *scare*, are members of different roleSetTypes. Derivatively we may speak of *classes of verbs* according to the *verb lexemes* represented in these classes of lexvals.

We name such classes of lexvals or verbs after the verb lexeme of the alphabetically first lexval where the roleSetType is found, for instance as in `abonnere`: <*agent*, *theme*> . This way of naming classes resembles a bit what is done for 'verb classes' in VerbNet. But note that in VerbNet 'verb class' is constituted by a combination of semantic and syntactic features, where the semantic features comprise not only roles but also logical form and elements of conceptual semantics, whereas our classes are defined by roles alone, hence the name roleSetTypes.

The number of members in each roleSetType by their rank is shown in Figure 1 and shows a Zipfian distribution. The cumulative distribution is shown in Figure 2. The three most common, `abonnere` ('subscribe'): <*agent*, *theme*>, `abbreviere` ('abbreviate'): <*agent*, *patient*> and `abdisere` ('abdicate'): <*agent*>, occur in in respectively 3,163, 2,344 and 1,307 lexvals. The first of these classes can broadly be described as representing agentive, bivalent verbs whose second argument does not undergo a change of state, as in (8).

(8) de hamstrer matvarer
they hoard foodstuffs
'they hoard foodstuffs'

The second most common roleSetType represents agentive bivalent verbs whose second arguments are internally changed – the referent of the object of the verb *abbreviere* ('abbreviate') is made shorter. The third class is the class of agentive intransitives, e.g., *abdisere* ('abdicate').

On the tail end of the frequency list, there are 12 roleSetTypes with 3 members each, 31 with 2 and 70 with 1. The reason for the large number of roleSetTypes with one member is found in the source syntactic annotation. The roleSetTypes `trives` <*experiencer*, *location*>, for example, represents one verb: *trives* 'thrive', annotated for location (9).

(9) deltagerne trives her
participants.def thrive here
'the participants are thriving here'

The number of rare roleSetTypes follows from the NorVal tagging, which for *trives* is *intrObl-oblLoc*: an intransitive verb that selects for an oblique locative. Of 30 verbs with this syntactic tag in NorVal, *trives* is the only one that takes an experiencer subject. Note crucially that the verb *trives*, without a locative, is also a member of the larger roleSetType `ane` <experiencer>, with 94 members, among them *lide* 'suffer' and *koble av* 'relax'. The large number of classes, then, needs to be seen in relation with the syntactic tagging of arguments given in NorVal.

5.2. Class granularity

As already said, our annotation yields a database where separate verbs are semantically tagged only for semantic roles. This contrast with the English VerbNet, where verbs such as *hold* and *neglect*, although annotated using the same semantic roles, belong to different classes based on semantic definitions: the class `hold-15.1` is defined semantically as *contact*, while `neglect-75.1` is defined as ¬*handle*. Our annotation thereby results in larger classes – verbs that would belong to different classes in a semantically richer classification, end up in the same class. Verbs like *antenne* 'ignite' and *vie* 'marry', for example, both end up in `abbreviere` <agent, patient>.

For the purpose of using the database as a lexical resource for UD graphs, the low semantic granularity is not an issue. The current stage of

semantic role	freq.	semantic role	freq.
agent	10,691	*topic*	946
theme	6,792	*recipient*	600
patient	3,376	*destination*	570
null-role	1,667	*orientation*	521
experiencer	1,226	*pivot*	451
stimulus	1,134	*formal*	399

Table 3: The 10 most frequent semantic roles

the database, however, is a suitable point of departure for adding more detailed semantic definitions, as is done in the English VerbNet, and for specifying valid syntactic alternations for different verb frames. *vie*, for example, may be followed by the segment <*til* co-patient> 'marry x to y'. This is not possible for *antenne*. As a syntactico-semantic resource, syntactic subcategorization of the semantic classes stands out as a central future endeavour for creating a full Norwegian VerbNet. However, the current stage of the database provides ample opportunities for examining syntactico-semantic phenomena. Some key statistics and possible usage domains are given below.

The distribution of the 10 most frequently annotated semantic roles is given in Table 3. *null-role* is annotated for reflexive pronouns lacking semantic participant status. The role *formal* represents syntactically required but semantically vacuous pronouns, as found for instance with weather-related verbs (Bolinger, 1973).

In total, 31,351 semantic role tokens were annotated for 18,830 sense-distinct lexvals (i.e., among the 17,200 lexvals in NorVal, the syntactic frame in many cases hosts more than one sense in terms of semantic roles, bringing the number of role-annotated lexvals up to 18,830). On average, each lexval frame contains 1.7 semantic roles (1.9 if null-roles and formal subjects are not counted).

The database allows for queries about the co-occurrence of semantic roles in Norwegian: out of a total of 6,792 instances of the role *theme*, 141 are followed by a *co-theme*, tentatively illustrating the structural frequency of themes co-occuring with an equally salient undergoer. Out of a total of 1,342 instances of the role *experiencer*, 1,020 co-occur in structures with a stimulus, 666 of which precede the experiencer role (10) and 354 of which surface after the experiencer (11).

(10) vi$_{stim}$ avskrekker villsvinene$_{exp}$
we scare.off boars.def
'we scare off the boars'

(11) vi$_{exp}$ frykter [at huset bygges]$_{stim}$
we fear that house build.pass
'we fear the building of the house'

The database further has the potential to be used for research in lexical semantics, for example for the question of what kind of verbs combine with formal subjects in Norwegian compared to other Germanic languages. A query that looks for formal subjects *formal* followed a *results* role yields a semantic structure in Norwegian (12) that is not found for English in the English Verbnet.

(12) det$_{formal}$ slår om til [å regne]$_{result}$
it changes to rain
'it is (the weather) changing to rain'

5.3. Semantic roles and UD

As described in section 3.4, the NorVal frames were mapped to UD frames, and the semantic roles were aligned with UD functions as was shown in Table 2. In general, the mapping from VerbNet to UD is many-to-one – different semantic functions maps to a single syntactic annotation. For example, both benefactive objects (*he defended* ***them***) and objects of verbs of breaking (*she destroyed* ***the vase***) reduce to a single UD relation *obj*.

However, we also find – albeit to a lesser extent – one-to-many mappings from semantics to syntax. This is because semantic rolesets that are annotated for the same role are distinguished into multiple syntactic frames based on whether the semantic role is represented by a clausal or nominal element. The two semantically identical objects in (13) are assigned different syntactic relations in UD, respectively *obj* and *ccomp*.

(13) I accepted {it / that they wrote novels}

The 250 semantic classes (i.e., roleSetTypes) map to 63 UD configurations at the syntactic level. The most common UD configuration is *nsubj-obj* – structures with a nominal subject and object – with 7,226 roleSet tokens. The second most common is the class of argument structures with a single nominal argument – *nsubj* – with 2,602 member frames.

The mapping from semantic frames to syntactic structures is an overall reductive process. Looking at single frames, however, these often increase. Both of the semantic frames <*agent theme*> and <*experiencer stimulus*>, when following the syntactic conventions in UD, map to five syntactic frames: *nsubj-advcl*, *nsubj-ccomp*, *nsubj-obj*, *nsubj-obl* and *nsubj-xcomp*. The verb *frykte* 'fear' selects for three of the syntactic structures (14), while the phrasal verb *fortvile over* 'despair about' showcases the remaining two (15).

(14) vi$_{nsubj}$ frykter {dem$_{obj}$ / [at huset
we fear them that house.def
bygges]$_{ccomp}$ / [å tape]$_{xcomp}$}
build.pass to lose

'we fear them / that the house is built / to lose'

(15) han$_{nsubj}$ fortviler over {[vår skjebne]$_{obl}$
he despairs about our destiny
/ [hva som må gjøre]$_{advcl}$}
what that must done.pass
'he despairs about our destiny / what must be done'

6. Conclusion/Outlook

We have presented NoVRol, a semantic role lexicon of Norwegian verbs. We started from the NorVal valency lexicon and identified the semantic roles that the verbs in this database assign to their arguments, based on the VerbNet annotation guidelines. In the next step, we encoded the verbs' valency frames in UD, allowing for an easy mapping from UD functions to semantic roles that could be used, e.g., in semantic role labeling of running text.

Going beyond the current annotation, we believe there are also several exiciting avenues for further development of the resource. Integrating the detailed syntactic information from the NorVal frames, ideally in the same format as in the English VerbNet, would enable the creation of a much more detailed verb class system. This would make cross-linguistic studies of argument structure easier given the common annotation framework. Such a resource could in turn enable more research into regularities in the syntax-semantics mapping. Moreover, it would then also be possible to create detailed semantic representations of event structure. This could be exploited in semantic parsing, which was indeed the motivating application for our work.

Data availability

The dataset is available at `https://github.com/Universal-NLU/NoVRol` under the CC BY-SA 4.0 license.

Lasha Abzianidze, Johannes Bjerva, Kilian Evang, Hessel Haagsma, Rik van Noord, Pierre Ludmann, Duc-Duy Nguyen, and Johan Bos. 2017. The Parallel Meaning Bank: Towards a multilingual corpus of translations annotated with compositional meaning representations. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 242–247, Valencia, Spain. Association for Computational Linguistics.

Dorothee Beermann and Lars Hellan. 2004. A treatment of directionals in two implemented hpsg grammars. In *Proceedings of the HPSG04 Conference*. CSLI Stanford.

Dwight Bolinger. 1973. Ambient it is meaningful too. *Journal of Linguistics*, 9(2):261–270.

Johan Bos. 2013. The Groningen meaning bank. In *Proceedings of the Joint Symposium on Semantic Processing. Textual Inference and Structures in Corpora*, page 2, Trento, Italy.

Long Chen, Zhihong Jiang, Jun Xiao, and Wei Liu. 2021. Human-like controllable image captioning with verb-specific semantic roles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16846–16856.

Ann Copestake. 2002. *Implementing Typed Feature Structure Grammars*. CSLI Publications.

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

Charles J Fillmore. 1968. *The Case for Case*, pages 1–88. Holt, Rinehart, and Winston, New York.

Jamie Y. Findlay, Saeedeh Salimifar, Ahmet Yıldırım, and Dag T. T. Haug. 2023. Rule-based semantic interpretation for Universal Dependencies. In *Proceedings of the Sixth Workshop on Universal Dependencies (UDW, GURT/SyntaxFest 2023)*, pages 47–57, Washington, D.C. Association for Computational Linguistics.

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.

Lars Hellan. 2019. Construction-based compositional grammar. *Journal of Logic Language and Information*, 28:101–130.

Lars Hellan. 2022. A valence catalogue for norwegian. In *Natural Language Processing in Artificial Intelligence*, pages 49–104, Cham. Springer International Publishing.

Lars Hellan. 2023. A unified cluster of valence resources. In *Logic and Algorithms in Computational Linguistics 2021*, pages 311–347, Cham. Springer International Publishing.

Lars Hellan and Tore Bruland. 2015. A cluster of applications around a deep grammar. In *Proceedings from The Language Technology Conference, LTC2015, Poznan*, pages 503–508.

Jena D. Hwang and Martha Palmer. 2015. Identification of caused motion construction. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, pages 51–60, Denver, Colorado. Association for Computational Linguistics.

Beth Levin. 1993. *English verb classes and alternations: A preliminary investigation*. University of Chicago press.

Olga Majewska and Anna Korhonen. 2023. Verb classification across languages. *Annual Review of Linguistics*, 9(1):313–333.

Jaouad Mousser. 2010. A large coverage verb taxonomy for Arabic. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Stephanie Needham and Ida Toivonen. 2011. Derived arguments. In *Proceedings of the LFG11 Conference*, pages 401–421. CSLI Stanford.

Wessel Poelman, Rik van Noord, and Johan Bos. 2022. Transparent semantic parsing with Universal Dependencies using graph transformations. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4186–4192, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Quentin Pradet, Laurence Danlos, and Gaël de Chalendar. 2014. Adapting verbnet to french using existing resources. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*.

Siva Reddy, Oscar Täckström, Slav Petrov, Mark Steedman, and Mirella Lapata. 2017. Universal semantic parsing. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 89–101, Copenhagen, Denmark. Association for Computational Linguistics.

Arka Sadhu, Tanmay Gupta, Mark Yatskar, Ram Nevatia, and Aniruddha Kembhavi. 2021. Visual semantic role labeling for video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5589–5600.

Karin Kipper Schuler. 2005. *VerbNet: A broad-coverage, comprehensive verb lexicon*. Ph.D. thesis, University of Pennsylvania.

Lucien Tesniere. 1959. *Éléments de syntaxe structurale*. Klincksieck, Paris.

Expanding Russian PropBank: Challenges and Insights for Developing New SRL Resources

Skatje Myers[*], **Roman Khamov**[*], **Adam Pollins**[*], **Rebekah Tozier**[*],
Olga Babko-Malaya[†], **Martha Palmer**[*]
[*]University of Colorado
{firstname.lastname}@colorado.edu

[†]BAE Systems
olga.babko-malaya@baesystems.com

Abstract

Semantic role labeling (SRL) resources, such as Proposition Bank (PropBank), provide useful input to downstream applications. In this paper we present some challenges and insights we learned while expanding the previously developed Russian PropBank. This new effort involved annotation and adjudication of *all* predicates within a subset of the prior work in order to provide a test corpus for future applications. We discuss a number of new issues that arose while developing our PropBank for Russian as well as our solutions. Framing issues include: distinguishing between morphological processes that warrant new frames, differentiating between modal verbs and predicate verbs, and maintaining accurate representations of a given language's semantics. Annotation issues include disagreements derived from variability in Universal Dependency parses and semantic ambiguity within the text. Finally, we demonstrate how Russian sentence structures reveal inherent limitations to PropBank's ability to capture semantic data. These discussions should prove useful to anyone developing a PropBank or similar SRL resources for a new language.

Keywords: Semantic role labeling, Semantically annotated resources, Russian semantics

1. Introduction

The ability to identify the semantic elements of a sentence (*who* did *what* to *whom*, *where* and *when*) is crucial for machine understanding of natural language and downstream tasks such as information extraction (MacAvaney et al., 2017), question-answering systems (Yih et al., 2016), text summarization (Mohamed and Oussalah, 2019), and machine translation (Rapp, 2022). The process of automatically identifying and classifying the predicates in a sentence and the arguments that relate to them is called semantic role labeling (SRL).

Using the PropBank schema (Palmer et al., 2005) (Pradhan et al., 2022), a Russian-language lexicon and corpus was manually annotated, called Russian PropBank (which we will refer to as RuPB1) (Moeller et al., 2020). In this paper, we present our work expanding RuPB1 (we refer to the expanded version as RuPB2), the challenges encountered, and our proposed solutions. We present this discussion to benefit future work for new PropBanks and semantic representations in other languages, many of which may encounter similar challenges during annotation and in representing the semantics of target languages.

In particular, we have been creating new frames [1] and expanding double-annotated and adjudicated coverage of the verbs, as well as expanding the scope of annotation to include participles and both relativizers and their head words. Our efforts have resulted in a smaller but more thorough dataset. This paper first provides a general overview of our project's source material and goals as well as related projects that facilitated the process in Section 2. Next, we distinguish the respective scopes of RuPB1 and RuPB2 in Section 3. Section 4 covers changes made to RuPB1's frames and the issues faced when adding frames to RuPB2. We provide an overview of our infrastructure and annotation process in Section 5. In Section 6, we discuss sources of disagreement between annotators and the guidelines we devised to resolve them. Finally, we review how Russian's dropped copulas provide a challenge for accurate, detailed semantic representation in Section 7.

2. Background

Proposition Bank (PropBank) takes a verb-oriented but very generalizable approach to representing semantics. The list of permissible semantic roles is defined by the sense of each verb using numbered labels, ARG0 through ARG6. Typically an ARG0 is similar to a Proto-agent (per Dowty (1991)), and is the Agent or Experiencer, while ARG1 is usually the Patient or Theme of the predicate, similarly to a Proto-patient. By

[1] https://github.com/cu-clear/RussianPropbank/

5th Workshop on Designing Meaning Representations (DMR 2024) @LREC-COLING-2024, pages 30–38
21 May, 2024.

generalising the arguments in this way, automatic semantic role labelers can produce useful information even if they misidentify the frame. Additionally, there are adjunct-like arguments, called argument modifiers (ARGM), to incorporate other semantically relevant information such as location (ARGM-LOC) and direction (ARGM-DIR).

The standard approach to developing a PropBank for a new language is to begin by defining a valency lexicon, known as a set of PropBank Frame Files, that defines the predicate-argument structure for all predicates to be annotated (Xue and Palmer, 2009; Zaghouani et al., 2010; Palmer et al., 2006; Bhatt et al., 2009). Once a sufficient number of frames has been defined, the annotation process begins, with the annotators referring to the frames for guidance for each individual predicate. In order to maintain complete annotation coverage for each sentence, additional frames are typically added during the annotation process. Double-blind annotation is recommended, followed by an expert adjudication pass. It is expected that the annotation process will reveal various ways in the which the original frame definitions need to be revised, sometimes resulting in follow-on revisions to previous annotations.

The Low Resource Languages for Emergent Incidents (LORELEI) project[2] sought to explore techniques for rapidly developing natural language processing technologies for low-resource languages. The dataset released as part of this project consists of parallel corpora for 23 low resource languages across many genres, such as newswire, phrasebooks, and weblogs. A subset of the English data was manually annotated with PropBank SRL.

The RuPB1 corpus (Moeller et al., 2020) project constructed 364 frames and annotated PropBank-style semantic roles on a portion of the Russian *newswire* and *phrasebook* sentences that paralleled the English dataset. This consists of 91 *newswire* sentences (2,228 tokens) and 496 *phrasebook* sentences (2,471 tokens). The previous work focused on annotating high-frequency verbs, which resulted in most sentences in the corpus having partial annotation. Our work has focused on filling in missing predicate annotations to produce fully labeled sentences in order to facilitate use of this corpus for training and testing SRL models and for evaluating how well annotation projection methods, such as those used by the Universal PropBanks project (UPB) (Jindal et al., 2022), map to SRL designed for the target language. The latter requires fully-annotated sentences to determine which predicates have been missed, added, or misplaced by the projection. See Table 1 for more details.

	RuPB1	RuPB2
# frames	364	497
# sentences	587	257
# predicates	431	331

Table 1: Comparison of annotation coverage between the partial annotation of RuPB1 and the smaller but completely annotated sentences of RuPB2.

Russian PropBank is not the only resource for Russian SRL. Russian FrameBank (Lyashevskaya and Kashkin, 2015) is a project to develop FrameNet-style (Baker et al., 1998) frames designed for Russian and annotate examples of those frames from the Russian National Corpus [3]. Their annotation scheme uses 96 distinct semantic roles, such as Result or Beneficiary, organised in an hierarchical graph. Frames for approximately 4,000 target verbs, adjectives, and nouns were constructed, and over 50,000 examples of these frames were annotated. There is a fundamental difference in the approach of both resources: Russian FrameBank is rooted more in lexical semantics, while PropBanks are more focused on the syntax-semantics interface (Levin, 1993). As a result, RuPB offers a coarser-grained, more general SRL schema. Instead of having 96 specific semantic roles, PropBank uses the numbered arguments described above. For instance, an ARG0 can be either an Agent or an Experiencer depending on the predicate. Additionally, while FrameNet accounts for peripheral arguments and modifiers, Russian FrameBank does not; its annotations focus only on the core arguments of a given example predicate. In contrast, for each predicate, RuPB labels both the core arguments and modifiers (PropBank's equivalent of peripheral arguments). Additionally, RuPB2's goal is to annotate every predicate in a given sentence, instead of only annotating a specific, example predicate. Unfortunately, this means there is no automatic way for RuPB to take advantage of the 50,000 annotated example sentences in Russian FrameBank without extensive manual review, since the latter's annotations only provide partial coverage of the predicates in a sentence. For the same reason, Russian FrameBank does not provide an appropriate evaluation corpus for UPB.

3. Scope

As discussed above, while RuPB1 prioritised developing frames in the order of verb frequency, our

[2] https://www.darpa.mil/program/low-resource-languages-for-emergent-incidents

[3] https://ruscorpora.ru/

aim was to ensure that sentences have *complete* annotations, so that this resource can also be used as a test dataset. As a result, RuPB2 produced 200 sentences of *phrasebook* and 57 sentences of *newswire* with all predicates annotated.

Besides the additional verb annotation, we also extended the scope of RuPB from only verbs to include participles for 36 verbs and 9 relative-head pairs, such as обнаруженных ‘discovered’. These are annotated with the same frames that they would be as verbs (обнаружить ‘to discover’).

RuPB2 also expanded annotations to include R-ARGs, in alignment with EnPB guidelines. Previously, the relativizer was the only argument annotated (such as который ‘who’ in this example):

(1)	мальчик	который	любит	кошек
	mal’čik	*kotoryj*	*ljubit*	*košek*
	boy	who	loves	cats
	-	ARG0	pred	ARG1

RuPB2 now captures both the relativizer and the head noun:

(2)	мальчик	который	любит	кошек
	mal’čik	*kotoryj*	*ljubit*	*košek*
	boy	who	loves	cats
	ARG0	R-ARG0	pred	ARG1

Our scope is still narrower than than of the current EnPB, which extensively annotated nominalizations and predicative adjectives. Some of these additional parts of speech may be added to RuPB in the future, such as nominalizations and eventive nouns, depending on applications. Some of the eventive nouns were added to EnPB during projects that focused on disasters, such as tornado.01, which captures arguments for things such as death toll and Fujita scale.

Another type of predication that EnPB includes, but RuPB does not, are adjectival predicates, such as blue.01: “He was blue from the cold.”

4. Framing

As discussed above, the development of a high-quality, comprehensive valency lexicon is the cornerstone of the PropBanking process. Thanks to RuPB1, we began with a pre-existing set of Russian Frame Files. Our goal with RuPB2 was twofold: 1) to add enough frames to get full sentence coverage; 2) and to expand the scope of the predicates being annotated.

In addition to the expansion, 134 new frames were added, and many previous frames were reexamined. During the initial stages of RuPB2, we ran into the issue of using different terms when discussing framing decisions, and settled on the following clarifications for the terms: alias, roleset, and predicates.

An alias is a grammatical or syntactic form of a verb. Both *drank* and *drunk* are aliases of the verb *drink*. A roleset is a particular sense of a verb as well as a list of its core arguments according to their semantic roles. Rolesets also include all aliases of the verb in question.

Roleset id: drink.01	
‘ingest liquid’	
ARG0	drinker, agent
ARG1	liquid
ARG2	source of liquid

Table 2: Roleset for drink.01

Predicates are collections of rolesets. Many verbs are polysemous, and each sense or meaning of the verb (predicate) is captured by different rolesets. The predicate *drink* can have two rolesets, *drink*.01 and *drink*.02, as in ’I drank water from a well’ vs ’I drink to your health’. See Tables 2 and 3.

Roleset id: drink.02	
‘salute’	
ARG0	drinker, agent
ARG1	thing saluted

Table 3: Roleset for drink.02

Determining whether a given token warrants its own predicate or roleset, or is simply an alias of an existing roleset, can be challenging, especially in morphologically rich languages. For additional examples and details of our framing process, please refer to the RuPB2 Framing Guidelines on the website.

As discussed by Moeller et al. (2020), Russian verbs can undergo many morphological processes that sometimes change the verb’s aspect but can change semantic meaning as well.

For example, the reflexive affix -ся can simply change a verb’s grammar (new alias) but can also add a new sense (new roleset). The verb молить, ‘to beg’ (Table 4), becomes ‘to pray’, молиться (Table 5), when the reflexive affix is added. By comparison, хотеть and хотеться, ‘to want’, have no semantic difference.

Often these kinds of differences lead to discussions of semantic domains and analysing frequencies of arguments in the literature. The RuPB2

Roleset id: молить.01	
molit’ ‘to beg’	
ARG0	asker, agent
ARG1	person being begged
ARG2	thing asked for

Table 4: Roleset for молить.01.

Roleset id: молиться.01	
molit'sja 'to pray'	
ARG0	pray-er, agent
ARG1	prayer
ARG2	deity

Table 5: Roleset for молиться.01.

guidelines err on the side of making different rolesets as opposed to different aliases when the framer is unsure. This can be referred to as *splitting* as opposed to *lumping*.[4] There are two main factors that led to this decision. Firstly, having a clear golden rule speeds up the process of creating new frames. Secondly, it means an ill-judged decision is easily reversible. Should a framer make two separate rolesets instead of separate aliases, it is always easier to go back through annotations and deterministically merge two different tags into a single tag. This is simple to implement and much easier than the reverse: deciding that two aliases should be separate rolesets and manually re-annotating every occurrence according to the new senses.

In contrast with EnPB, other PropBank projects have set a precedent of splitting over lumping, as seen in the Turkish PropBank (Ak et al., 2018), where very rich morphological processes result in lots of very similar rolesets. Verbs with negative or modal affixes are given their own frames despite having identical rolesets.

Our suggestion is that any potential PropBank should have explicit guidelines on splitting vs. lumping. On the one hand, a liberal approach to splitting may result in the amount of frames ballooning drastically. Yet a conservative approach may result in much time and effort spent on reversing previous decisions. Both linguistic and computational factors must be considered.

In the case of RuPB2, aside from some minor edits to existing frames (such as typos and confusing example sentences), there were a few decisions that resulted in different annotations compared to RuPB1. Some involved removing frames entirely.

The first case was the frame мочь.03 'can, may', which has no core arguments. This differs from мочь.01 'can, have ability', which has an ARG1 (*agent with ability*) and an ARG2 (*ability itself*).

мочь.03 can be seen in the following sentence in Figure 1:

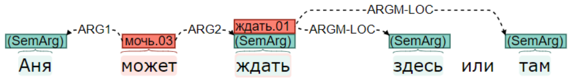


Figure 1: Anna can wait here or there

In RuPB1, может would have been marked as мочь.03. This sense is contrasted with a sentence such as Figure 2:

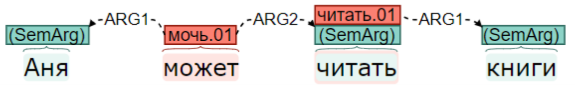


Figure 2: Anna can read books

In Figure 2, annotators should mark может as мочь.01, with ARG1 being 'Anna', and ARG2 being 'read'. In RuPB2, the может should be marked as ARGM-MOD for 'wait' not as its own predicate, since it is a modal indicating possibility (see Figure 3). In RuPB2, мочь.03 is removed entirely.

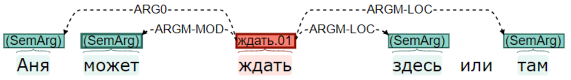


Figure 3: Anna can wait here or there

Likewise, we removed the roleset давай(те).09, which can be translated into English as 'let's', as in "let's look at a few examples". Instead of having a dedicated roleset, the verb will simply be marked ARGM-MOD, since it is essentially a hortative, modal verb.

Although we initially added быть.08, which was modeled on EnPB be.03 (the auxiliary verb 'will/was/were'), we eventually opted to remove this frame.

(3) Мы будем есть
My budem est'
We will eat.

EnPB set out to annotate semantic components including temporal relations as an ARGM (Kingsbury and Palmer, 2002). One could argue that быть.08 should be included in RuPB2 to adhere more closely to its English counterpart. Ultimately, быть.08 seemed to perform more of a functional, placeholder role; annotators would label this sense to avoid confusion with other быть senses. Because быть.08 lacks significant lexical information, we opted for its discontinuation.

5. Annotation Process

All RuPB2 annotation and adjudication was completed through the text-annotation platform INCEpTION (Klie et al., 2018). INCEpTION's interface streamlines corpus creation, annotation, and adjudication (the INCEpTION term for this phase is 'curation'). Our project required an environment

[4] Splitting and lumping have long been used by lexicographers to illustrate a bias in favor of either more coarse-grained senses or more fine-grained senses.

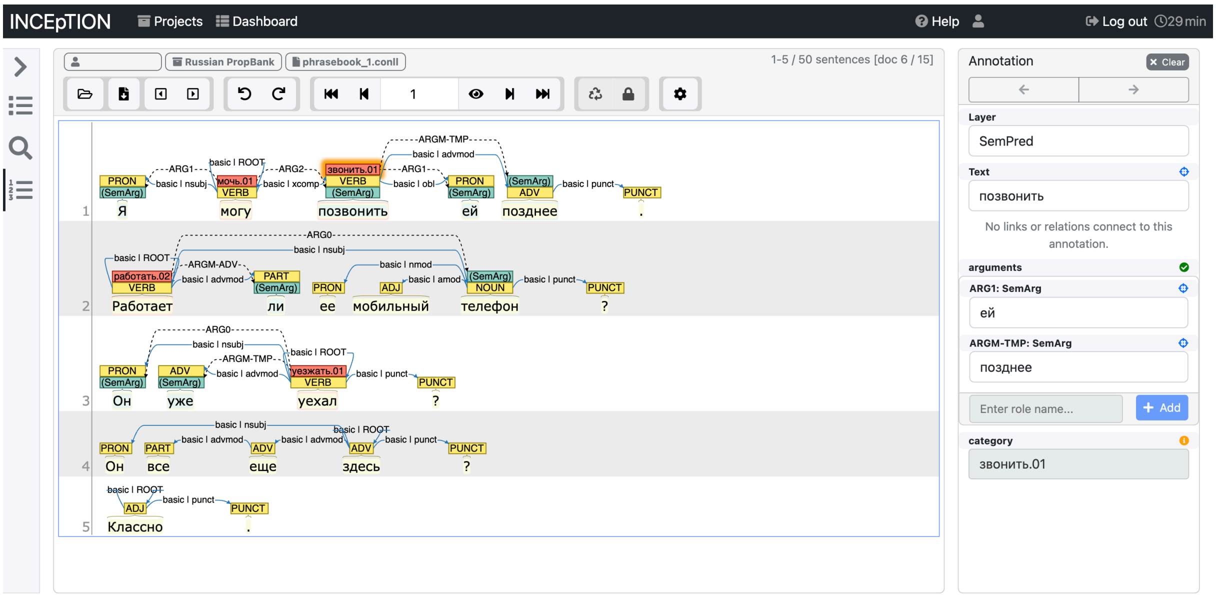

INCEpTION
Projects
Dashboard
Help
Log out
29 min
Russian PropBank
phrasebook_1.conll
1-5 / 50 sentences [doc 6 / 15]
1
Я мочь.01 могу звонить.01 позвонить ей позднее .
PRON (SemArg) VERB VERB (SemArg) PRON (SemArg) (SemArg) ADV PUNCT
ARG1 basic I nsubj basic I ROOT ARG2 basic I xcomp ARGM-TMP basic I advmod ARG1 basic I obl basic I punct
2
работать.02 Работает ли ее мобильный телефон ?
VERB PART (SemArg) PRON ADJ (SemArg) NOUN PUNCT
ARG0 basic I nsubj basic I ROOT ARGM-ADV basic I advmod basic I nmod basic I amod basic I punct
3
Он уже уезжать.01 уехал ?
PRON (SemArg) ADV (SemArg) VERB PUNCT
ARG0 basic I nsubj ARGM-TMP basic I advmod basic I ROOT basic I punct
4
Он все еще здесь ?
PRON PART ADV ADV PUNCT
basic I nsubj basic I advmod basic I advmod basic I ROOT basic I punct
5
Классно .
ADJ PUNCT
basic I ROOT basic I punct
Annotation
Clear
Layer
SemPred
Text
позвонить
No links or relations connect to this annotation.
arguments
ARG1: SemArg
ей
ARGM-TMP: SemArg
позднее
Enter role name...
Add
category
звонить.01

Figure 4: INCEpTION Annotation mode.

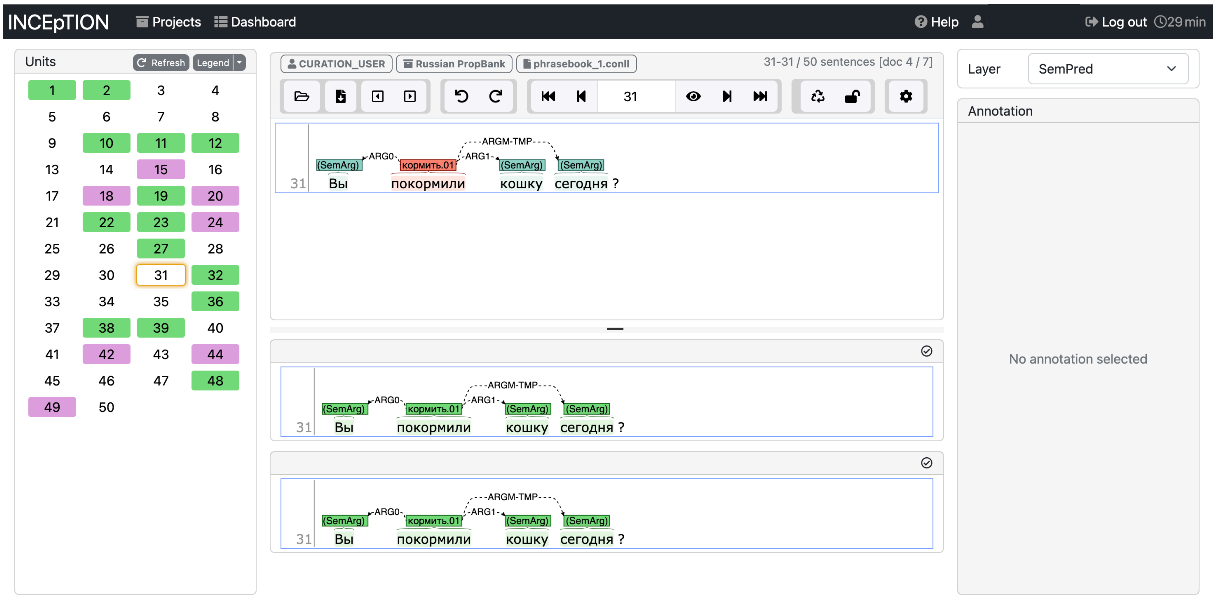


Figure 5: INCEpTION Curation mode.

that would allow multiple users to annotate a semantic layer of predicates and arguments. This annotation process was additionally assisted by being able to simultaneously view dependency parse and part of speech layers. We automatically parsed the sentences using UDPipe (Straka and Straková, 2017) and provided these parses as our initial data in INCEpTION.

Figure 4 provides an example of the RuPB2 sentences using the annotation feature of the INCEpTION platform. Looking more closely at sentences 4 and 5 in Figure 4, these are examples of INCEpTION feature layers before RuPB2 annotation was complete. No semantic roles could be annotated for sentences 4 and 5 due to the lack of predicates and arguments. The figure displays the layers that assisted the annotators: the sentences are written in Cyrillic text and further organized by each word's part of speech (yellow boxes). In addition, each sentence is syntactically parsed (e.g., subjects, objects, and sentence roots). In contrast with 4 and 5, sentences 1 through 3 have semantic predicates (red boxes) and arguments (various ARG arrows and green SemArg boxes) annotated. Observe that each predicate takes a specific verb frame, such as мочь.01, and core (numbered) arguments are distinguished from ARGMs (e.g., adverbial, ARG-ADV or temporal, ARG-TMP) When clicking on a predicate, a pane on the right of the platform also shows the details of the semantic predicate layer.

Upon completion of a document, annotators submitted their finished work to an adjudicator for adjudication (the 'curation' pass). The adjudicator compared the finished annotations between users to assess inter-annotator agreement, with discrepancies highlighted. Additionally, the adjudication process resulted in the adjudicator creating a final, gold standard, fully annotated sentence. This process is illustrated in Figure 5; the top sentence reflects the adjudicator's gold standard annotated sentence, whereas the lower two sentences are the annotators'. For simplicity, only semantic predicate and argument layers are shown in Figure 5, but the layers included in Figure 4 are also available during the adjudication process.

Since the annotation guidelines and framing decisions evolved concurrently with the annotation process itself, all members of the RuPB2 group participated in the adjudication step. This thorough process allowed all aspects, framing, annotation challenges, and future work to be discussed. Annotation challenges and future work are presented more thoroughly in the following sections.

6. Annotation Challenges

RuPB annotators rely on an underlying Universal Dependencies (UD) syntactic parse to resolve ambiguity (de Marneffe et al., 2021). This parse itself sometimes introduces new ambiguity. Unlike EnPB annotators, who tag arguments as spans of words, RuPB annotators must identify and tag the word that corresponds to the argument's head. The automatic UD parser's choice of head is not always intuitive or consistent, and we observed it caused annotator disagreement most frequently in part-whole constructions and phrases that comprise more than one temporal modifier. Phrases containing locative modifiers were another source of disagreement. The counts of these phenomena that occurred in the RuPB2 sentences are totaled in Table 6, largely occurring in the more complex *newswire* sentences.

Pseudopartitives	7
Temporal Doublets	4
Locative Modifiers	4

Table 6: Cases of Challenging Annotation

6.1. Part-Whole Constructions

The head of a quantified nominal phrase is usually the inner nominal, which refers to the whole entity quantified (e.g., две тысячи **людей** / *two thousand **people***). By contrast, the head of a partitive construction is the outer nominal or part (***tons** of rice*). When the parser labels a quantifier as a noun instead of a numeral, the quantifier becomes the head of that phrase (**тысячи** людей / ***thousands** of people*) because the construction appears syntactically partitive. The part of speech of the quantifier thus changes the head of the phrase in the parse, though it does not affect the phrase's lexical meaning.

These constructions are *pseudopartitives*, and should not be analyzed as having the same syntax as partitives (Falco and Zamparelli, 2019). Their prevalence varies from one language to another, but they appear more frequently in Russian UD parses than in English. Compare English *a million **residents*** (numeral) and ***millions** of residents* (pseudopartitive) with Russian **миллион** жителей lit. 'million of residents' and **миллионы** жителей 'millions of residents' (both pseudopartitive). Annotators must take care to choose the head of each argument when working with a dependency parse that does not handle constructions such as pseudopartitives.

6.2. Temporal Doublets

Temporal modifiers occasionally appear in a series. However, the parser does not always treat the modifiers either as a single oblique nominal or as two separate ones, as seen in Figure 6. At first, annotators tagged according to the parse, but then noticed these inconsistencies and needed a different solution. In EnPB practice, arguments that comprise conjuncts are treated as a single argument and never tagged twice. With this in mind, we chose to treat these constructions as asyndetic coordination, in which the first element is the head per UD guidelines.[5]

6.3. Locative Modifiers

Annotators encountered difficulty as they decided whether to tag locative modifiers as arguments of the verb or to consider them as modifying a noun and thus not tagged. Straightforward cases of both the former type ('People died **in the village**') and the latter ('The head of the program **in Bangladesh** expressed his fears') appeared, in which annotators agreed.

Yet in ambiguous cases, annotators diverged. For example, in тысячи людей **в Индии и Бан-**

[5] https://universaldependencies.org/u/dep/conj.html

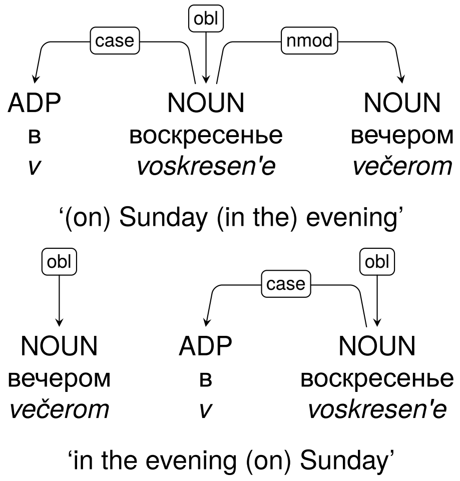


Figure 6: Two temporal modifier arguments, identical save word order, with arbitrarily different parses

гладеше до сих пор обращаются … 'Thousands of people **in India and Bangladesh** are still seeking …', they did not agree as to whether this locative was an argument of 'seeking' or modified 'people'. We resolved this by tagging the locative as a modifier of the verb in each ambiguous case, as we preferred to be thorough and prevent omissions.

7. Implicit Predicates

Russian usually drops the present tense linking verb, есть.[6] A lexical unit is omitted in instances similar to those in English where *be* would appear as *am*, *is* or *are*. For instance, one does not say "I am a student" in Russian but literally я студентка "I student." Ten percent of the sentences in our dataset were affected by dropped copulas (i.e., sentences had one less predicate or were completely unable to be annotated), predominately in the *phrasebook* portion.

This issue of dropping copulas is not limited to Russian; the World Atlas of Language Structures Online (Stassen, 2013) reports that 45% of the languages accounted for in their database (175 of 386 languages) allow zero copula constructions with nominal predicates.

8. Conclusion

We have presented numerous issues that were encountered during our endeavor to expand and complete PropBank annotation for RuPB2, a Russian SRL dataset for training and testing purposes. In the future, RuPB2 can be further expanded to include nominalizations and light verb constructions to provide better coverage. We described our approach to constructing frames for Russian, which can provide a precedent for other morphologically rich languages and others with similar characteristics. More particularly, we analyzed the complexity in differentiating between predicates and modal verbs. We discussed our solutions to frequent cases of annotator disagreement, as well as the importance of the parse in settling ambiguities. In the final section, we discussed the challenges of implicit predicates that can be found in zero-copula sentences, which we expand on below. Throughout the development of RuPB2, there has been an aim to stay true to this schema and maintain parity with EnPB, all the while reflecting the semantics of Russian with the highest accuracy possible. These discussions should prove useful to anyone building a new PropBank for another language.

Although there are many benefits of the PropBank schema, it is important to also consider limitations when constructing a semantic corpus for a new language. PropBank can capture shallow semantic information about who did what to whom, but a deeper complete sentence representation that includes discourse relations and modality can be more effective. Uniform Meaning Representations (UMRs), (Van Gysel et al., 2021), provide a cross-lingual approach to such a representation. UMRs are based on the popular Abstract Meaning Representations project (Banarescu et al., 2013) which directly incorporates English PropBank for predicate argument structures. The ability of AMR-UMR to represent implicit predications yields a strategy for capturing semantics that is not covered by PropBank alone. Our Russian PropBank provides an essential foundational element for this type of richer, more nuanced semantics. The discussion and suggestions for how to develop guidelines and frame files for a Slavic language that are contained in this paper should provide a road-map for anyone else undertaking such an endeavor.

9. Acknowledgements

We gratefully acknowledge the support of DARPA HR0011516904-Lorelei, Semantic Annotation and Technology Transfer, a subaward from LDC, DARPA FA8750-18-2-0016-AIDA – RAMFIS: Representations of vectors and Abstract Meanings for Information Synthesis, a sub award from RPI on DARPA KAIROS Program No. FA8750-19-2-1004, and NSF 1764048 RI: Medium: Collaborative Research: Developing a Uniform Meaning Representation for Natural Language Processing. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the

[6] The phenomenon where a subject and predicate are not overtly connected through a linking verb is known as a zero (or null) copula.

authors and do not necessarily reflect the views of NSF, DARPA or the U.S. government.

10. Bibliographical References

References

K. Ak, C. Toprak, V. Esgel, and O. T. Yildiz. 2018. Construction of a turkish proposition bank. *Turkish Journal of Electrical Engineering and Computer Sciences*, 26(1):570–581.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *COLING 1998 Volume 1: The 17th International Conference on Computational Linguistics*.

L. Banarescu, Claire Bonial, Shu Cai, M. Georgescu, K. Griffitt, U. Hermjakob, K. Knight, P. Koehn, Martha Palmer, and N. Schneider. 2013. Abstract meaning representation for sembanking. *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186.

Rajesh Bhatt, Bhuvana Narasimhan, Martha Palmer, Owen Rambow, Dipti Sharma, and Fei Xia. 2009. A multi-representational and multi-layered treebank for Hindi/Urdu. In *Proceedings of the Third Linguistic Annotation Workshop (LAW III)*, pages 186–189, Suntec, Singapore. Association for Computational Linguistics.

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

David Dowty. 1991. Thematic proto-roles and argument selection. *language*, 67(3):547–619.

Michelangelo Falco and Roberto Zamparelli. 2019. Partitives and partitivity. *Glossa: a journal of general linguistics*, 4(1).

Ishan Jindal, Alexandre Rademaker, Michał Ulewicz, Ha Linh, Huyen Nguyen, Khoi-Nguyen Tran, Huaiyu Zhu, and Yunyao Li. 2022. Universal Proposition Bank 2.0. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1700–1711, Marseille, France. European Language Resources Association.

Paul Kingsbury and Martha Palmer. 2002. From TreeBank to PropBank. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*, Las Palmas, Canary Islands - Spain. European Language Resources Association (ELRA).

Jan-Christoph Klie, Michael Bugert, Beto Boullosa, Richard Eckart de Castilho, and Iryna Gurevych. 2018. The inception platform: Machine-assisted and knowledge-oriented interactive annotation. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 5–9. Association for Computational Linguistics. Event Title: The 27th International Conference on Computational Linguistics (COLING 2018).

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

Olga Lyashevskaya and Egor Kashkin. 2015. FrameBank: A database of russian lexical constructions. In *Communications in Computer and Information Science*, pages 350–360. Springer International Publishing.

Sean MacAvaney, Arman Cohan, and Nazli Goharian. 2017. GUIR at SemEval-2017 task 12: A framework for cross-domain clinical temporal information extraction. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1024–1029, Vancouver, Canada. Association for Computational Linguistics.

Sarah Moeller, Irina Wagner, Martha Palmer, Kathryn Conger, and Skatje Myers. 2020. The Russian PropBank. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5995–6002, Marseille, France. European Language Resources Association.

Muhidin Mohamed and Mourad Oussalah. 2019. Srl-esa-textsum: A text summarization approach based on semantic role labeling and explicit semantic analysis. *Information Processing & Management*, 56(4):1356–1372.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106.

Martha Palmer, Shijong Ryu, Jinyoung Choi, Sinwon Yoon, and Yeongmi Jeon. 2006. Korean PropBank. *LDC Catalog No.: LDC2006T03 ISBN*, pages 1–58563.

Sameer Pradhan, Julia Bonn, Skatje Myers, Kathryn Conger, Tim O'gorman, James Gung, Kristin Wright-bettner, and Martha Palmer. 2022. PropBank comes of Age—Larger, smarter, and more diverse. In *Proceedings of the 11th Joint Conference on Lexical and Computational Semantics*, pages 278–288, Seattle, Washington. Association for Computational Linguistics.

Reinhard Rapp. 2022. Using semantic role labeling to improve neural machine translation. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3079–3083, Marseille, France. European Language Resources Association.

Leon Stassen. 2013. Zero copula for predicate nominals (v2020.3). In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Zenodo.

Milan Straka and Jana Straková. 2017. Tokenizing, pos tagging, lemmatizing and parsing ud 2.0 with udpipe. In *Proceedings of the CoNLL 2017 shared task: Multilingual parsing from raw text to universal dependencies*, pages 88–99.

Jens EL Van Gysel, Meagan Vigus, Jayeol Chun, Kenneth Lai, Sarah Moeller, Jiarui Yao, Tim O'Gorman, Andrew Cowell, William Croft, Chu-Ren Huang, et al. 2021. Designing a uniform meaning representation for natural language processing. *KI-Künstliche Intelligenz*, 35(3-4):343–360.

Nianwen Xue and Martha Palmer. 2009. Adding semantic roles to the Chinese TreeBank. *Natural Language Engineering*, 15(1):143–172.

Wen-tau Yih, Matthew Richardson, Chris Meek, Ming-Wei Chang, and Jina Suh. 2016. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 201–206, Berlin, Germany. Association for Computational Linguistics.

Wajdi Zaghouani, Mona Diab, Aous Mansouri, Sameer Pradhan, and Martha Palmer. 2010. The revised arabic propbank. In *Proceedings of the Fourth Linguistic Annotation Workshop*, pages 222–226. Association for Computational Linguistics.

Unveiling Semantic Information in Sentence Embeddings

Leixin Zhang[2=], David Burian[1=], Vojtěch John[1], Ondřej Bojar[1]

[1]Faculty of Mathematics and Physics, Charles University
[2]University of Tübingen, Germany
leixin.zh@gmail.com, david.burian@me.com,
vojtik.john@seznam.cz, bojar@ufal.mff.cuni.cz

Abstract

This study evaluates the extent to which semantic information is preserved within sentence embeddings generated from state-of-art sentence embedding models: SBERT and LaBSE. Specifically, we analyzed 13 semantic attributes in sentence embeddings. Our findings indicate that some semantic features (such as tense-related classes) can be decoded from the representation of sentence embeddings. Additionally, we discover the limitation of the current sentence embedding models: inferring meaning beyond the lexical level has proven to be difficult.

Keywords: sentence embedding, transformation vector, semantic information

1. Introduction

Word embeddings have frequently been used as input in deep neural networks. Sentence embeddings are supposed to encapsulate sentence meanings into vectors. However, representing an entire sentence as a vector of fixed length poses significant challenges. Obtaining sentence embeddings is not as straightforward as extracting word embeddings based on contextual information from text. Embeddings merely based on surrounding text can be less representative at the sentence level.

Additionally, evaluating the quality of sentence embeddings or assessing whether these embeddings effectively encapsulate the meanings of sentences often requires a human-annotated corpus with well-defined semantic categories or sentence similarity scores.

In this study, we convert Czech sentences in the COSTRA dataset into sentence embeddings using SBERT and LaBSE models. COSTRA dataset (Barančíková and Bojar, 2020) is a collection of Czech sentences with semantic labels. Each set consists of a 'seed' sentence and transformation sentences that are derived from the seeds. The objective of this study is to assess whether sentence embeddings trained by SBERT and LaBSE retain semantic information and whether vectors in the same transformation class (with some similarity in semantics) show affinity in high dimensional space, which is tested by using clustering and classification algorithms to investigate whether vectors from the same class can be distinguished from vectors of other classes in high dimensional space.

The content of our paper is structured as follows: Section 3 presents a detailed introduction to the COSTRA dataset and an overview of our evaluation methods. In Section 4, we implement the dimension reduction technique to visualize sentence embeddings in 2D graphs. Section 5 attempts to predict new sentence embeddings with extracted transformation vectors. Section 6 implements cluster separation tests to assess within-class cohesion and between-class separation for 13 transformation classes. In Section 7, supervised methods are employed to train and predict transformation labels. Finally, Section 8 compares the results in all evaluation tasks and discusses the separability of transformation vectors.

2. Previous Studies

In this section, we introduce previous research on sentence embeddings, as well as the evaluation methods employed for assessing sentence embeddings.

2.1. Previous Studies on Sentence Embeddings

Word embeddings represent word meanings in space, and sentence embeddings are supposed to encapsulate sentence meanings into vectors, ideally of fixed lengths. There are two approaches to generating sentence embeddings. One is unsupervised learning of sentence embeddings. For instance, Yang et al. (2018) and Arora et al.

[=] Authors with equal contribution.

21 May, 2024.

Class	**Description**	**Example** (Translated from Czech)
seed	original sentence	*Four members of my family lost their lives.*
ban	negative imperative	*Four members of my family cannot lose their lives!*
possibility	possibility modality	*Four members of my family probably lost their lives.*
past	past tense	*In those days, four members of my family lost their lives.*
future	future tense	*Four members of my family will one day lose their lives*.
opposite meaning	opposite sense	*Four members of my family were born*.
generalization	make it more general	*Four people died.*
minimal change	minimal alteration	*Four members of that family lost their lives.*
nonsense	by shuffling words	*Life lost members of my family.*
different meaning	by shuffling words	*Four members of my family lost a member.*
formal sentence	a more formal style	*Four members of my family closed their eyes forever.*
simple sentence	a simplistic style	*Four people of my family died.*
nonstandard	a colloquial style	*Almost my whole family died there.*
paraphrase	paraphrase	*Four of my relatives died.*

Table 1: Seed and Transformation classes in COSTRA

(2019) proposed an unsupervised method to construct sentence embeddings. They calculate the weighted sum of word embeddings[1] and then remove principal components to enhance embedding quality.

Nevertheless, the dominant method in prior research for generating sentence embeddings is supervised learning towards the relations (e.g. natural language inference, Conneau et al., 2017) we want to get from the embeddings.

The sequence-to-sequence architecture was used to generate sentence embeddings in machine translation tasks, with the encoder's output serving as the sentence representation. LASER (Artetxe and Schwenk, 2019) is an instance. It is a multilingual LSTM-based encoder-decoder model trained on parallel corpora across 93 languages (Goswami et al., 2021). However, it is challenged due to the suboptimal semantic representation. Reimers and Gurevych (2020) state that LASER fails in assessing the similarity of sentence pairs, despite its good performance in identifying exact translations.

More recently, transformer and BERT-based models have received increased attention. SBERT (Reimers and Gurevych, 2019) stands as a state-of-the-art model for generating sentence embeddings (Ham and Kim, 2021). Multilingual models have also been studied in recent years. Reimers and Gurevych (2020) fine-tune the monolingual SBERT model (Reimers and Gurevych, 2019) with a parallel corpus that includes 50 languages and leveraged knowledge distillations. Chidambaram et al. (2019) propose mUSE (Multilingual Universal Sentence Encoder), trained on parallel data in 16 languages. LaBSE (Feng et al., 2022) is another multilingual BERT-based model, trained on a dual encoder with 6 billion sentence translation pairs across 109 languages. These three multilingual models have demonstrated strong performance in previous studies (Devine et al., 2021; Reimers and Gurevych, 2020; Ham and Kim, 2021). In our study, we use SBERT and LaBSE, two models that support the Czech language to generate sentence embeddings.

2.2. Sentence Embedding Evaluation

The evaluation of sentence embeddings in previous studies includes linguistic probing tests, semantic similarity tests, and other downstream classification tests (Conneau and Kiela, 2018).

Linguistic probing tasks start with investigating surface information, like decoding sentence lengths or assessing whether the original words can be detected from a sentence embedding (Adi et al., 2016). The syntactic evaluation examines whether sentence embeddings can detect neighbouring word shifts, part of speech tags, coordination inversion, number or gender agreement, depth of the syntactic tree, etc. (Perone et al., 2018; Pimentel et al., 2020; Hupkes et al., 2018). Other downstream classification tasks involve sentiment analysis and opinion polarity (Perone et al., 2018, Conneau et al., 2018).

The semantic similarity test is also popular in sentence embedding evaluation. Models are assessed by computing the correlation between the human-labeled similarity scores of sentence pairs and the model-predicted distance (e.g. cosine dis-

[1] The actual deep learning tasks in which the word embeddings obtained can vary, such as autoregressive (e.g. LSTM) or non-autoregressive language modelling.

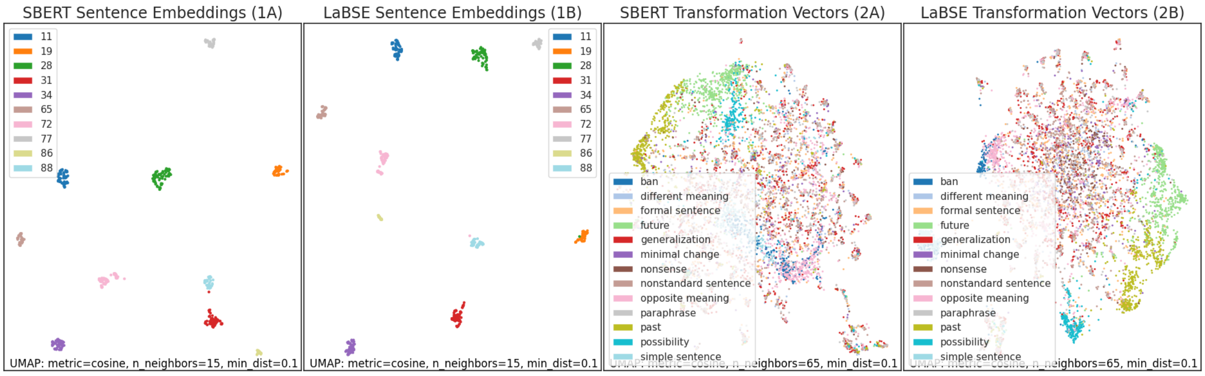


Figure 1: Visualization of sentence embeddings (1A) & (1B) and transformation vectors (2A) & (2B). (1A) & (1B) illustrate sentence embeddings of 10 randomly selected seeds and their corresponding transformed sentences. Each set of a seed sentence and its derived sentences is indicated by a seed index and represented with a distinct colour.

tance) of two sentence embeddings.

However, many semantic studies on sentence embeddings often fall short in providing insights into instances where models consistently underperform. Our research adopts a novel approach, potentially serving as a controlled experiment. By maintaining consistency in the seed sentences' information while altering only specific features in 13 classes, our research offers advantages in examining embedding transformations in detail.

3. Dataset and Sentence Embeddings

COSTRA (Barančíková and Bojar, 2020) is the evaluation dataset in our study. It comprises 6,968 Czech sentences, out of which 126 are seed sentences. The remaining sentences are transformation sentences derived from the seed sentences. These transformation sentences are categorized into 13 classes. Table 1 presents the descriptions of the 13 transformation classes and example sentences translated from the Czech COSTRA dataset.

In our study, we use SBERT[2] and LaBSE, two multilingual models with Czech language support to generate sentence embeddings. We differentiate two types of vectors: sentence embeddings and transformation vectors. **Sentence embeddings** are generated directly from SBERT and LaBSE models. **Transformation vectors** are vectors with their corresponding seed embeddings subtracted, in order to remove additional information from the seed sentence. In other words, given a transformed e.g. generalized sentence (with its embedding denoted as *generalization*$_i$ for short), we also consider the corresponding seed sentence (with the seed embedding denoted as *seed*$_i$) The transformation vector of this sentence pair is represented as *generalization*$_i$ - *seed*$_i$.

In the following sections, we aim to study whether transformation vectors in one class demonstrate a clustering tendency (within class cohesion) and whether they can be distinguished from transformation classes of other types (between-class separation).

4. Dimension Reduction and Visualization

This section presents a preliminary study of sentence embeddings and transformation vectors through dimension reduction and visualization. UMAP (Uniform Manifold Approximation and Projection) (McInnes et al., 2018) was employed as our dimension reduction technique and visualization tool.[3]

Firstly, we explore the spatial distribution of the sentence embeddings. Our assumption is that a seed sentence, sharing more identical words with its derived sentences, may lead to closer proximity to its transformed sentences than sentences belonging to other seed sets. To test the hypothesis, we randomly visualize 10 seed sentences along with sentences that are derived from them. Secondly, our analysis aims to explore whether transformation vectors (obtained by subtracting seed embeddings from their sentence embeddings) within the same transformation class (e.g. future transformation vectors) tend to group together.

Sentence embeddings from SBERT and LaBSE are depicted in Figure 1 (1A) & (1B). Each set

[2] To produce SBERT Sentence embeddings we used pre-trained multilingual model '*paraphrase-multilingual-MiniLM-L12-v2*'.

[3] PCA and T-SNE are also tested in the initial experiments, while the performance is much worse than UMAP, thus not presented in the paper.

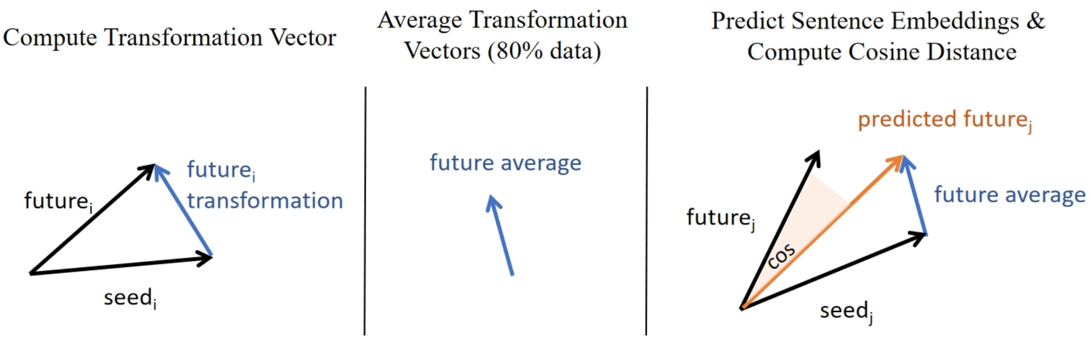

Compute Transformation Vector
future$_i$
future$_i$ transformation
seed$_i$
Average Transformation Vectors (80% data)
future average
Predict Sentence Embeddings & Compute Cosine Distance
predicted future$_j$
future average
future$_j$
cos
seed$_j$

Figure 2: Cosine Similarity Computation between True and Predicted Sentence Embeddings

of sentence embeddings (the seed sentence and sentences derived from it) generally forms a cluster, suggesting that sentences tend to be situated close to their seed sentences.

In the results in Figure 1 (2A) & (2B), the tendency of the transformation vectors of the same class clustering together is observed only for certain classes, particularly tense-related classes ('past' and 'future'). Some classes form a cluster with only a part of the sentences, such as 'opposite meaning' and 'simple sentences'. However, transformation vectors of other classes (e.g. 'nonstandard sentence' and 'generalization') are dispersed across the space.

Additionally, it is worth noting that despite the different model architectures, and different lengths/dimensions of sentence embeddings of SBERT and LaBSE, their visualization results after the dimension reduction display comparable behaviour.

5. Predictive Capacity of Transformation Vectors

Section 4 demonstrates that transformation vectors in some (though not all) transformation classes are grouped together after dimension reduction. This section further evaluates the potential of transformation vectors to predict other sentence embeddings based on their seed embeddings. We assume the following property holds for transformation vectors: given a future-tense transformation vector (future$_i$ - seed$_i$), and the embedding of a different seed (seed$_j$), we can predict the embedding future_sentence$_j$ (sentence of its future tense) using Equation 1.

$$\mathrm{future_j} = \mathrm{future_i} - \mathrm{seed_i} + \mathrm{seed_j} \qquad (1)$$

In the actual experiment, 80% of the sentences in each class are used to extract transformation vectors. We compute the average of the 80% transformation vectors to predict the sentence embeddings for the remaining 20% of the sentences (as shown in the illustration in Figure 2). The quality of transformation vectors is assessed using the cosine similarity between the predicted sentence embeddings and the true sentence embeddings.

class	SBERT	LaBSE
possibility	0.94	0.95
past	0.93	0.91
future	0.92	0.91
different meaning	0.91	0.91
nonsense	0.90	0.90
formal sentence	0.88	0.88
minimal change	0.87	0.92
ban	0.85	0.91
paraphrase	0.82	0.81
nonstandard sentence	0.81	0.82
simple sentence	0.81	0.79
opposite meaning	0.75	0.83
generalization	0.70	0.66

Table 2: Cosine Similarity of predicted embeddings and true derivation sentence embeddings

5.1. Cosine Distance between Predicted and True Embeddings

The results in Table 2 show that the majority of the transformation classes have a cosine similarity score above 0.8. These findings imply that a number of predicted vectors lie close to their true sentence embeddings, especially those in 'possibility' and 'past' classes, both with very high scores.

However, in contrast, the 'generalization' class exhibits the lowest score (0.70 in SBERT and 0.66 in LaBSE), falling below the baseline (ranging from 0.72 to 0.78), obtained by using the same dataset but with shuffled transformation labels within each seed set.

This could be attributed to the varying degrees of transformation when a seed sentence is transformed into multiple generalization forms. If the transformation vectors do not align in a consistent vector direction, relying on the average of 80% of the vectors is inaccurate in predicting sentence embeddings. It is also worth mentioning that the cosine distance of the baseline with shuffled transformation labels reaches 0.72, suggesting that the embeddings of any arbitrary sentence and the arbitrary transformation of the sentence are close to each other.

5.2. Cosine Distance across Classes

To deal with the aforementioned challenge of varying transformation degrees within a class and the limitation of assessing transformation vectors solely relying on cosine distance from their true embeddings, we extend our assessment to the cosine distance of predicted sentence embeddings with actual embeddings across 13 classes.

Our underlying assumption is that although transformation vectors with varying degrees might not exhibit a consistent vector direction in space, transformation vectors in one class may still be restricted within a region that is distinguishable from the regions of other transformation classes. As a result, predicted sentence embeddings should show the highest cosine similarity with sentence embeddings of the target class, compared to those from other classes. For example, the sentence embedding predicted by the 'generalization' transformation vector, is compared with the true embedding *generalization*$_i$ (with the assumed highest cosine similarity), as well as with sentence embeddings of other classes derived from *seed*$_i$, such as *past*$_i$, *ban*$_i$, *nonsense*$_i$, etc. (with an assumed lower cosine similarity).

Figure 3 displays the results of the comparison across classes. Each row is normalized using min-max normalization. Darker hues indicate closer to 1, while lighter hues indicate scores near 0. We call it normalized predictability score, measuring how well the embeddings of the target classes are predicted from the transformation vectors of the source class.

The results suggest that the diagonal cells typically get the darkest hue and the remaining cells in the same row often display lighter shades. It implies a generally higher cosine similarity between the predicted embeddings and the actual embeddings of the target class compared to embeddings of other classes. In particular, the sentence embedding of 'ban' is the best-predicted class, although its cosine similarity score discussed in Section 5.1 does not rank high among the 13 classes.

However, the predictability varies across transformation classes. In the results of SBERT, the predictions of four classes ('different meaning', 'minimal change', 'non-sense', and 'paraphrase') display the highest cosine similarity scores with embeddings in a different class. For instance, the predicted embeddings of 'different meaning' show the highest cosine similarity with 'minimal change' embeddings, while the predicted 'non-sense' embeddings correlate most strongly with the true embeddings of 'different meaning'. Additionally, the cosine similarity values of the 'formal sentence' and 'simple sentence' classes are not sufficiently distinguished from the values of other classes.

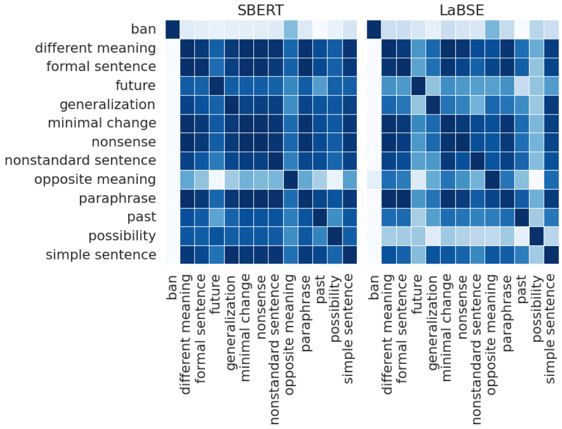


Figure 3: Cosine similarity between true and predicted embeddings. (Each row is normalized with min-max normalization. Darker hues indicate scores closer to 1, while lighter hues indicate scores near 0.)

We note that LaBSE outperforms SBERT in this experiment. There is only one instance of incongruence: predicted 'paraphrase' embeddings exhibit the highest cosine similarity with sentence embeddings of 'different meaning'. The generally better performance of LaBSE can also be observed in Figure 3.

6. Cluster Separation Test

This section analyzes whether the transformation vectors of the same class cluster together and are separated from other classes in space. We present a cluster separation test using the Calinski-Harabasz index.

$$\mathrm{CH} = \left[\frac{\sum_{k=1}^{K} n_k \|c_k - c\|^2}{K-1}\right] \Big/ \left[\frac{\sum_{k=1}^{K}\sum_{i=1}^{n_k} \|d_i - c_k\|^2}{N-K}\right] \qquad (2)$$

The Calinski-Harabasz index[4] (Equation 2) measures the ratio of between-cluster dispersion to

[4] K means the number of clusters; n_k is the number of points in k_{th} cluster; c_k represents the number of points and centroid of the k_{th} cluster; c is the global centroid; N is the total number of data points.

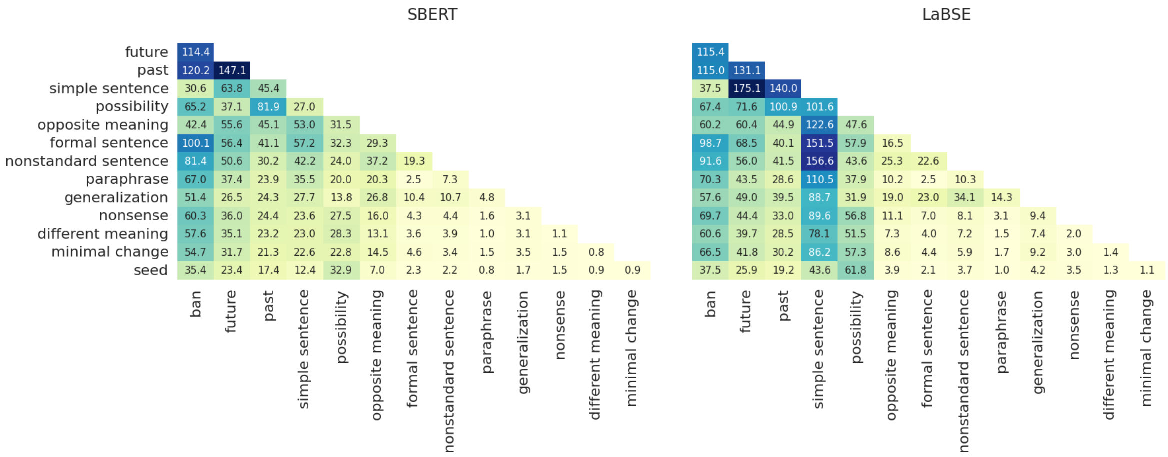


Figure 4: Pairwise Calinski-Harabasz index of transformation vectors from SBERT and LaBSE.

SBERT	LaBSE	mixSBERT	mixLaBSE
28.415	44.885	0.563	0.565

Table 3: Cluster separation test on 13 classes

inter-cluster dispersion. A higher value signifies well-separated clusters (Caliński and Harabasz, 1974).

In this study, we compute CH-Index in two ways. Firstly, we compare the performance of the two models by assessing transformation vectors in the 13 classes. Secondly, we conduct a pairwise test to assess the degree of separation of transformation classes in pairs.

We establish benchmarks for the CH Index by mixing up the transformation labels of the dataset. The CH index scores for 13 classes are shown in Table 3. LaBSE has a better performance than SBERT. Nevertheless, both models significantly outperform the baselines. Figure 4 presents the results of pairwise testing. Two baselines of mixed transformation labels have CH index values ranging from 0.392 to 0.899 for SBERT, and from 0.332 to 1.556 for LaBSE.

We observed that 'ban' and 'future' generally exhibit higher values, suggesting their better separation from other classes and within-class cohesion. In the results of LaBSE model, 'simple sentence' is the class with the highest CH-index scores, followed by 'ban', 'future' and 'possibility'. While for SBERT, the advantages of 'simple sentence' and 'possibility' classes are not observed. It indicates the discrepancies in the distribution patterns of transformation vectors in space obtained from SBERT and LaBSE.

Additionally, pairwise tests also show that other classes such as 'different meaning', 'minimal change' and 'paraphrase' often fall below the benchmark in both SBERT and LaBSE, suggesting insufficient separability of their transformation vectors in these classes.

7. Classification Task

In previous experiments, we utilized methods such as visualization, sentence embedding prediction, and clustering separation to assess the quality of transformation vectors from SBERT and LaBSE. This section introduces supervised methods to investigate whether transformation vectors can be decoded to predict transformation labels.

The classifiers used in our experiments consist of Random Forests, Support Vector Machine (SVM), and K-Nearest Neighbors (KNN). Depending on their unique strengths, these classifiers may decode transformation vectors in distinct ways. Random Forests use specific criteria and feature-based splitting to classify data (Breiman, 2001; Cutler et al., 2012). SVM has the ability to map inputs into high-dimensional spaces using the kernel trick (Schölkopf et al., 1999; Smola and Schölkopf, 2004). KNN adopts a local distance-based approach and assigns labels based on the known labels of neighbouring data points. We intend to investigate the potential of these diverse methods to extract semantic information (transformation labels) from transformation vectors.

In addition to the sentence embeddings from SBERT and LaBSE, we also generated TF-IDF weighted encoding of all vocabulary in COSTRA. The additional TF-IDF embeddings aim to assess the influence of lexical factors on classification performance. In other words, we aim to test whether certain words are unique to a particular transformation class, thereby potentially enhancing the prediction accuracy. Similarly to other tasks in our study, we use the mixed-up SBERT as our baseline.

The results in Figure 5 indicate high F1 scores

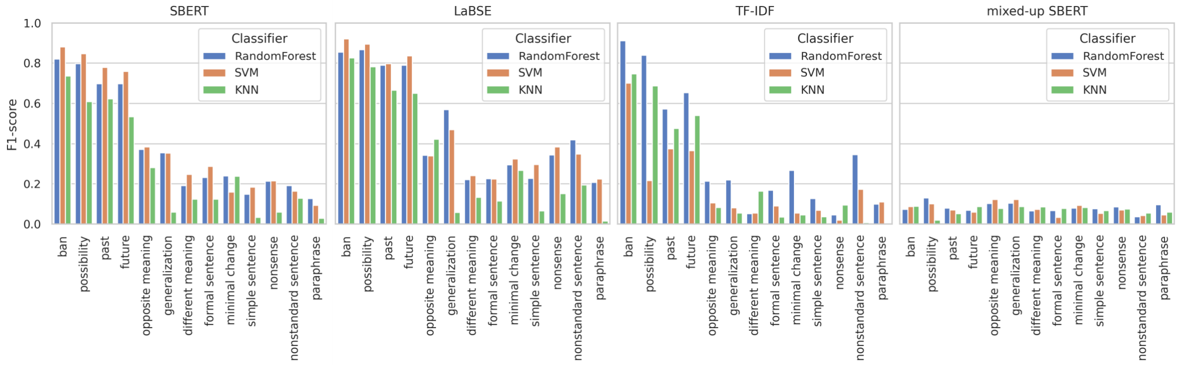


Figure 5: F1-scores for transformation label prediction

for four transformation classes: 'ban', 'possibility', 'past', and 'future'. The comparably high F1 score of TF-IDF embeddings suggests the substantial impact of the lexical factor on the predictability of these classes. In other words, sentences in these four classes tend to contain particular words that are unique to a class, contributing to their superior predictability.

Additionally, 'generalization' from LaBSE exhibits F1 scores higher than those of SBERT and TF-IDF. It on the one hand suggests that LaBSE outperforms SBERT in these two instances. On the other hand, it also implies that LaBSE may have a better ability to capture semantic information beyond the word level.

8. Discussion

In this section, we compare the results of the evaluation tasks implemented in our study and then discuss the separability of transformation vectors and to what extent the semantic features can be decoded from sentence embeddings.

8.1. Summary of Results in Evaluation Tasks

Transformation vectors in four transformation classes ('ban', 'possibility', 'past', and 'future') demonstrate good performance in almost all evaluation tasks: dimension reduction & visualization, sentence embedding prediction, cluster separation, and classification, and show consistent results in both models. This is in line with their pronounced separability from other classes. In contrast, some classes exhibit weak performance in almost all evaluation tasks, for instance, 'paraphrase', 'minimal change', 'formal sentence', and 'nonsense'.

Nevertheless, certain classes display varying performance across our four evaluation tasks and two models. For example, the LaBSE transformation vectors in the 'simple sentence' class excel in the sentence embedding prediction task (Figure 3) and the cluster separation test (Figure 4), but not in the classification task as shown in Figure 5.

The dimension reduction and visualization techniques may provide insight to speculate the reasons for such variations. Figure 1 displays that the clusters of the 'opposite meaning' and 'simple sentence' classes are formed only by some of the vectors in these two classes. The remaining data points within these two classes are dispersed throughout the space. This property (some data gathered together but some dispersed in space for a class) introduces complexity when assessing their separability with a single value in evaluation tests. Different evaluation methods may emphasize distinct properties of the vectors in a class and decode them in different manners. This could provide insight into the observed variations in performance for these classes across different evaluation tasks.

This analysis also suggests that while dimension reduction is criticized for the loss of information in high-dimensional spaces, it can instead offer supplementary insights when combined with visualization.

8.2. Separability Analysis

In the section above, we discussed that transformation vectors in some classes are not separable from others. It could be attributed to at least two factors. One factor is the inherent difficulty in distinguishing these classes from the rest, while the other factor is related to the limitations of the models themselves.

We notice that certain classes are inherently challenging to separate. For instance, sentences in the 'minimal change' class are less distinguishable from those in the 'different meaning' class. 'Paraphrase' is less distinguishable from 'simple sentence', 'formal sentence' and 'nonstandard sentence', simply because all of them are also a form of a paraphrase. The models' poor performance in evaluation tests may potentially correspond to the uncertainty inherent in human judgment. In other words, these classes might also pose difficulties in differentiation even for human assessors.

The second reason for weak performance in some tests lies in the models' limitations in capturing semantic information. For example, both models show relatively low prediction accuracy for 'nonsense' and 'opposite meaning' (with F1 for 'nonsense' < 0.4; 'opposite meaning' < 0.5), two types that are easy to detect for human assessors.

The good classification results of TF-IDF embeddings also reveal that the separability of classes can to a considerable extent stem from purely lexical factors. This observation suggests that inferring meaning beyond the lexical level is difficult for the two models, and sentence embeddings generated by current models lack a comprehensive representation of sentence meaning.

9. Conclusion

Our study analyzed sentence embeddings generated from two multilingual models: SBERT and LaBSE, evaluating using the Czech COSTRA dataset to test whether some semantic information is preserved and can be decoded from sentence embeddings.

Our visualization firstly demonstrates that transformation sentences are situated in proximity to their respective seed sentences in the vector space. To assess the semantic attributes of 13 transformation classes exemplified in the COSTRA dataset, we examined transformation vectors, obtained by subtracting seed embeddings from sentence embeddings to eliminate the original seed sentence information.

In addition to dimension reduction and visualization, we conducted three other evaluation tasks: sentence embedding prediction, cluster separation, and transformation label classification. Our findings indicate that both models exhibit comparable performance, with LaBSE slightly outperforming SBERT in certain evaluation tasks.

Furthermore, our analysis highlights that transformation vectors for some classes show better separability from other classes and reach better evaluation scores in evaluation tasks. However, the good outcome may be attributed to specific words that are exclusive to a particular class, as suggested by similarly good results obtained using simple TF-IDF. Although the lower performance observed in other transformation types may be due to their inherent difficulty in class detection, the limitations of the current models are not negligible: inferring meaning beyond the lexical level has proven to be challenging for them.

10. Acknowledgements

This work was partially supported by GAČR EXPRO grant NEUREM3 (19-26934X) and by the Grant Agency of Charles University in Prague (GAUK 244523).

11. Bibliographical References

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2016. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. *arXiv preprint arXiv:1608.04207*.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2019. A simple but tough-to-beat baseline for sentence embeddings. 5th International Conference on Learning Representations, ICLR 2017 ; Conference date: 24-04-2017 Through 26-04-2017.

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Petra Barančíková and Ondřej Bojar. 2020. Costra 1.1: An inquiry into geometric properties of sentence spaces. In *Text, Speech, and Dialogue: 23rd International Conference, TSD 2020, Brno, Czech Republic, September 8–11, 2020, Proceedings*, pages 135–143. Springer.

Leo Breiman. 2001. Random forests. *Machine learning*, 45:5–32.

Tadeusz Caliński and Jerzy Harabasz. 1974. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27.

Muthu Chidambaram, Yinfei Yang, Daniel Cer, Steve Yuan, Yunhsuan Sung, Brian Strope, and Ray Kurzweil. 2019. Learning cross-lingual sentence representations via a multi-task dual-encoder model. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 250–259, Florence, Italy. Association for Computational Linguistics.

Alexis Conneau and Douwe Kiela. 2018. SentEval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia. Association for Computational Linguistics.

Adele Cutler, D Richard Cutler, and John R Stevens. 2012. Random forests. *Ensemble machine learning: Methods and applications*, pages 157–175.

Peter Devine, Yun Sing Koh, and Kelly Blincoe. 2021. Evaluating unsupervised text embeddings on software user feedback. In *2021 IEEE 29th International Requirements Engineering Conference Workshops (REW)*, pages 87–95.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Koustava Goswami, Sourav Dutta, Haytham Assem, Theodorus Fransen, and John P. McCrae. 2021. Cross-lingual sentence embedding using multi-task learning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9099–9113, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jiyeon Ham and Eun-Sol Kim. 2021. Semantic alignment with calibrated similarity for multilingual sentence embedding. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1781–1791, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Dieuwke Hupkes, Sara Veldhoen, and Willem Zuidema. 2018. Visualisation and'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure. *Journal of Artificial Intelligence Research*, 61:907–926.

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Christian Samuel Perone, Roberto Silveira, and Thomas S. Paula. 2018. Evaluation of sentence embeddings in downstream and linguistic probing tasks. *ArXiv*, abs/1806.06259.

Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. 2020. Information-theoretic probing for linguistic structure. *arXiv preprint arXiv:2004.03061*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online. Association for Computational Linguistics.

Bernhard Schölkopf, Christopher JC Burges, Alexander J Smola, et al. 1999. *Advances in kernel methods: support vector learning*. MIT press.

Alex J Smola and Bernhard Schölkopf. 2004. A tutorial on support vector regression. *Statistics and computing*, 14:199–222.

Ziyi Yang, Chenguang Zhu, and Weizhu Chen. 2018. Parameter-free sentence embedding via orthogonal basis. In *Conference on Empirical Methods in Natural Language Processing*.

A Quantum Theory of Terms and New Challenges to Meaning Representation of *Quanterms*

Diego A. Burgos
Wake Forest University
1834 Wake Forest Road, Winston-Salem, NC 27109
burgosda@wfu.edu

Abstract

This article discusses the challenges to meaning representation of terms posed by a quantum theory of terms (QTT) that was recently reported. We first summarize this theory and then highlight the difficulties of representing *quanterms*, which is the name we coined for the view that the QTT has of terms as quantum systems by analogy with quantum objects in quantum mechanics. We briefly summarize the representation practices followed to date to record and represent terminology. We use findings reported in the literature to model both terms and *quanterms* and found that current representations of terms in specialized repositories are collapsed *quanterms* at the expense of other states of the original *quanterm*. In this work, both *quanterms* and collapsed *quanterms* are mathematically modelled following formulations used in quantum mechanics. These formulations suggest that representations of *quanterms* need to include information about the probabilities of *quanterm* states and the role they play in the entanglement of terms for phenomena such as specialized collocations.

Keywords: terminology, quantum theory of terms, meaning representation

1. Introduction

In terminology, a term is operatively defined as a conventional, non-compositional lexical unit linked to a meaning exclusively used in a specialized domain, e.g., medicine, architecture, etc. (Burgos & Vásquez 2024). Traditionally, mainstream terminology theories and models define the term as a bidimensional object (see, for example, ISO 704, 2013, pp. 36-37; Cabré, 1999, p. 35; Faber and L'Homme, 2022, p. 355) with the term and a linked concept or meaning as the two dimensions of this representation.

However, Burgos et al. (2024) recently reported a quantum theory of terms (QTT), which models the term as a dynamic, multidimensional object with the characteristics of a quantum system. *Quanterms*, as they could be called, challenge the representation models that have been so far used to represent terms and their meanings. The implications of this quantum model may have a significant impact in computational linguistics, language engineering, lexicography and terminography, terminology theory and other fields related to knowledge representation, understanding and generation.

This paper highlights these challenges in the light of the QTT. In order to attain this, we summarize the most common representations of the term that have been used to date. Then, we briefly introduce the QTT as well as an abstract representation of *quanterms*. This background helps pave the way for a discussion section about the challenges of operative meaning representation of *quanterms*. We close with some conclusions and ideas for forms of representation.

2. Representation of terms

One of the most widespread representations of terms is the lexicographic representation, that is, the definition of terms in specialized dictionaries. Likewise, this representation has been the starting point of other forms of representation (e.g., Adelstein 2007, p. 72; Mahecha & De Cesaris 2011; Berri, 2013; Burgos & Vásquez 2024). For example, the lexicographic definition is frequently turned into Pustejovsky's generative lexicon model (1995, 2011), which, in turn, uses feature structures akin to those proposed by Carpenter (1992) to represent lexicon entries based on meaning features. These structures have also been utilized in other frameworks such as unification grammars (see, for example, Francez & Wintner, 2011) or semantic theories (e.g., naive semantics, Dahlgren, 1988). Naturally, terms also are represented in terminological databases generally following an onomasiological philosophy. This basically means that each term has one single sense and that each database entry or record hosts only one concept or sense together with the term or terms that denote it (cf. WordNet, Fellbaum 1998). Specialized taxonomies or ontologies such as SNOMED CT follow a similar approach.

According to Burgos et al. (2024), what these representations have in common is that they are static and limited, like pictures of a particular state of the term. While we acknowledge the importance of the role played by these representations throughout the history of knowledge management and representation, we believe that a quantum view of the term, which we summarize below, calls for representation of terms reflecting the complexity of quantum systems.

3. Quantum theory of terms and *quanterms*

Burgos et al. (2024) view the term as a complex, multidimensional object with dynamic properties. This complexity is the result of a number of states and dimensions, in which the same term exists simultaneously. At the moment of observation or measurement, the term collapses into a particular state and updates or *freezes* a set of its properties according to the collapsed state. We will see below that this collapse may also happen due to the term's interaction with its environment because of a quantum phenomenon known as *decoherence*. The property of having several states at the same time is called superposition, which is described below.

3.1 Superposition of terms

It is this complex nature described above that motivates Burgos et al.'s Quantum Theory of Terms (QTT) by analogy of terms with instances of quantum superposition. Superposition in quantum mechanics describes an object that has several different simultaneous states (Miret 2015, p. 83). In the medical domain, this superposition was exemplified by Burgos (2024) with two instances of a medical condition, which were given two distinct denominations, namely, *alien hand* and *anarchic hand*. These two terms turn out to be not just simple variants, but they seem to be motivated by two states of the term, each with its own configuration of features in the conceptualization of the syndrome at two different observation moments. Thus, the first state and its denomination reflect the sensation that the hand belongs to another person, while the latter indicates that the hand appears to refuse to obey its owner.

Additional evidence was reported by Burgos and Vásquez (2024) based on an experiment with a language model in the form of word embeddings also in the clinical domain in Spanish. They observed that, while *alteration* is the prototypical semantic class for the term *mutation* in specialized repositories, the data show semantic class variation for the same term in the same domain. Two additional semantic classes were detected, namely *entity* and *process*. Each of the contexts in which each variant of *mutation* occurs makes a distinct observation in the dimension of conceptual variation with effects on the term's properties. It is interesting to note that this variation may also impact the agency of the unit, i.e., whether *mutation* semantically acts as experimenter or agent.

One interesting trait of quantum superposition is that some of the possible states of a quantum system may be mutually exclusive. This happens, not only with the two perceptions of *alien hand* and *anarchic hand*, but also with the case of *mutation* above*,* since entities and processes are mutually exclusive. This non-coexistence of feature values has a significant impact in the way these terms are represented using, for example, a concept tree of the domain.

The quantum superposition of terms suggests the existence of basic conceptual variants, i.e., variants that do not change into another concept, but rather undergo a change in some of the features of the same concept. Using *mutation* as an example, and assuming we could map each of its states and assign its features a numerical value, we would have a first graphic model of term superposition, that is, three observations or states of *mutation* as a *quanterm*, which we illustrate in Figure 1.

The figure shows three different states of *mutation* on the *z*-axis where the values of features 1, 3, 4, and 6 (e.g., part of speech, predicativity, composition, and form) remain constant across states, but the values of features 2, 4, and 7 (e.g., class, agency, and function) change depending on the moment of observation. Visually, this variation can be seen as a change in the color tones for changing features compared to the uniform tones of the stable features. Theoretically, the term in isolation simultaneously has a number of states whose properties can only be determined at the moment of observation. Thus, the model in Figure 1 represents that three states of *mutation* coexist in the conceptual dimension and that, in each of these states, its class, agency, and function can change depending on the moment and dimension the *quanterm* is observed.

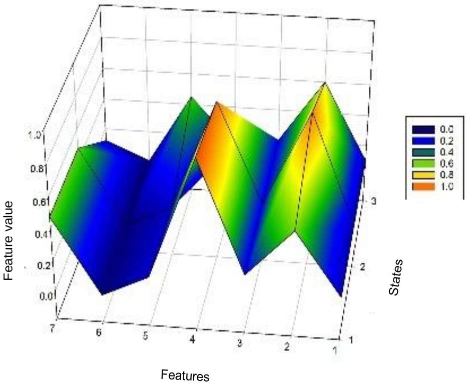


Figure 1. Model of *mutation* as a *quanterm* of three states

The model in Figure 1, however, can become more complex as the number of features, feature values, observations, and dimensions increase. The potential states of the *quanterm* in a more intricate scenario could therefore be represented by a matrix that combines these four factors. Regarding the possible number of dimensions, it is reasonable to think that it can always increase as more is known about the terminological phenomenon. However, we currently can predict six dimensions: dialect, level of specialization, social function, concept, domain, and time (see Table 1).

Superposition dimensions	
Time	Dialect
	Specialization
	Social function
	Concept
	Domain

Table 1: Dimensions where superposition can take place.

The reader may notice that these dimensions, except for *domain*, are related to a particular type of terminological variation reported in the literature (see Freixa, 2005). *Time* is an overarching dimension and accounts for diachronic aspects of the other five, including metaphorical phenomena. On the other hand, the *domain* dimension accounts for terms that can be at a crossroads between two or more domains or subdomains (e.g., *cell* in biology, veterinary science, and medicine). In each of these dimensions, the term can potentially take on a new feature or a different feature value in a particular observation.

As the *quanterm* becomes more complex, Figure 2 attempts to represent multiple states and features of a hypothetical *quanterm* in multiple dimensions.

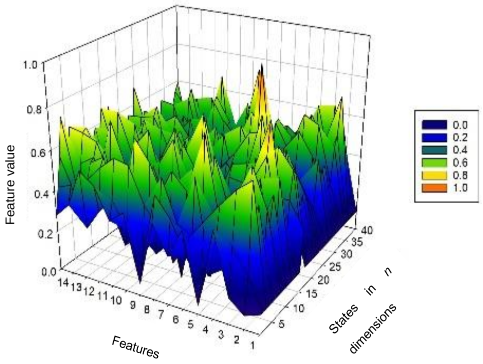


Figure 2. Hypothetical *quanterm*

For the model in Figure 2, we use a normal distribution of hypothetical feature values. It is a conservative representation based on the assumption that the *quanterm* is reasonably stable even though its features can be variable; otherwise, it would end up being a different concept. Visually, this stability can be seen as a lot of green and blue in the middle area of the graph with some peaks of color variations for significant changes in feature values that occur in particular states.

An example of the potential growth in the number features of a *quanterm* can be seen in *palatine tonsil*. The fact that that this term can be defined from a number of different subdisciplines increases the number of features that make up this *quanterm*. Depending on when and where the measurement of this term happens, its feature configuration would change. This occurs because a dentist, for example, gives prominence to features that may not be relevant to an anatomist, a pathologist, or a speech therapist, who would in turn highlight other features of the term when they use it in their respective domains while being the same *quanterm*.

A caveat is necessary here that the model in Figure 2 does not capture yet another layer of complexity added by the interaction or interdependence between *quanterms*. We describe such interaction below, which the QTT calls term entanglement.

3.2 Term entanglement

In quantum mechanics, entanglement refers to the interaction between particles such that the state of an object can be used to predict the state of another object (Miret 2015, p. 126). This property of *quanterms* allows for measuring the state of one term anticipating at the same time information about the state of other terms. A hypothetical example of this interaction can be the impact that a variation in the semantic class of a term in the conceptual dimension may have on, say, the agency of another term in the same dimension or in a different one. The QTT predicts that entanglement can happen even if the involved terms are far away from each other.

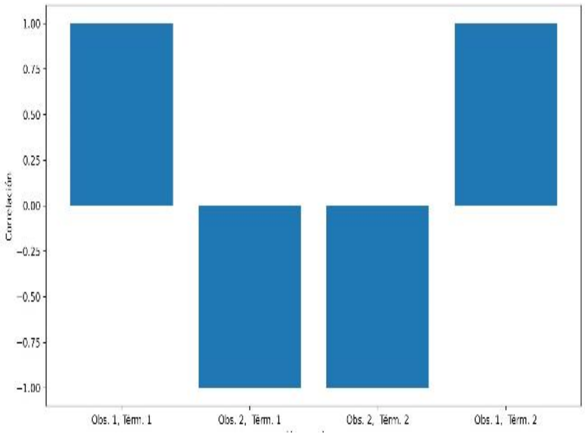


Figure 3. Entanglement of *quanterms*

Figure 3 illustrates possible effects of entanglement between two *quanterms*. The first observation of term 1 (Obs. 1, Term. 1) predicts that the state of Term 2 (Obs. 1, Term. 2) changes in a positive correlation. That is, if we assign numerical values to the features of Term 1, the values of Term 2 would change in the same direction. In the second observation of the same term, however, the correlation is negative. In other words, if the values of Term 1 increase, those of the other one decrease.

Besides other phenomena, entanglement of *quanterms* can explain specialized collocations. For example, a predicative term like *cancer* often selects terms referring to organs or tissues, such as *breast*, *prostate*, *stomach*, *skin*, etc., to produce collocations such as *breast cancer*, *gastric cancer*, *prostate cancer*, etc. Entanglement allows for predicting a correlation between the feature values of *cancer* and those of its collocation bases. Thus, we could reasonably anticipate that if *cancer* has a high value for the feature *alteration* (i.e., disease), the value of the feature *disease target* will proportionally increase in terms like *prostate* or *breast*. Let us use the definition of *prostate* in the Mosby Medical Dictionary (Villanueva et al. 1999) to clarify this point. *Prostate* is defined and anatomically described as a male gland, but there is no feature in its definition indicating that this gland is a target of cancer, perhaps because its value in this measurement of the term is very low. The feature *disease target*, however, is activated or its value increases in the domain dimension of oncology in positive correlation with the value of the feature *alteration* of terms like *cancer*. This entanglement, though, may not occur to the same extent for *prostate* in other domains such as anatomy or urology, which seems to be the state defined by the Mosby Dictionary above. The QTT attributes particular definitions or conceptualizations of terms to quantum decoherence, a phenomenon that is described below.

3.3 Decoherence

There are two reasons why a quantum system may collapse into one of its states, namely, the mere act of measuring it and its interaction with its environment. In quantum mechanics, this collapse is known as decoherence. The term is considered a *quanterm*, i.e., a quantum system, because, in isolation, it is in an indetermined number of different states at the same time, that is, it can be defined in multiple ways, even though some of those definitions may seem to conflict with each other. Any term in abstract, without any further definition or textual context, is a *quanterm*. Measuring a term may take the form of defining it or conceptualizing it, which involves determining the semantic features that delimit its specialized meaning. When this measuring operation takes place, the multiple states of the *quanterm* collapse into the meaning or conceptualization that it has been given, and it becomes a *collapsed term*, like the ones we currently see represented in dictionaries or ontologies.

Decoherence also happens as soon as the *quanterm* interacts with its environment, that is, with other terms and expressions in the context of a specialized text. The more specialized and specific the context, the more delimited its state is. We have an example of this explained above; the *quanterm* "*mutation"* collapses into a different state (entity, alteration, or process) depending on what environment it interacts with. It must be considered also that decoherence may sometimes be conditioned by a term entanglement.

4. *Quanterm* representation challenges

It is important to clarify that *quanterm* superposition does not refer to polysemy, but to the same term, concept or sense, which undergoes at a higher level a number of states (i.e., variations) at the same time, even if they are mutually exclusive, without changing into another concept. Polysemy has been successfully handled by lexicographic representations with a semasiological orientation (i.e., general dictionaries) and semantic networks, such as WordNet, as well as by formalisms based on qualia structures, such as the one in Figure 4 reported by Núñez Torres (2013).

[Door (x ∨ y)
QUASTR [FORM: physical_object´ (x), frame´ (y)
CONST: obstruction´ (x), aperture´ (y)
TELIC: BECOME closed´ / open´ (x), do´ (z, [go.through´ (z, y)])
AGENT: artifact´ (x ∨ y)]]

Figure 4. Representation of a collapsed state of the polysemous term *door*

Of a similar nature are onomasiological specialized resources such as terminological databases and ontologies. All these resources, however, always record collapsed states of *quanterms*. See, for example, the representation of the collapsed concept *myocardial infarction* in SNOMED CT in Figure 5[1], which, interestingly, seems to be also an instance of term entanglement, also known as a specialized collocation.

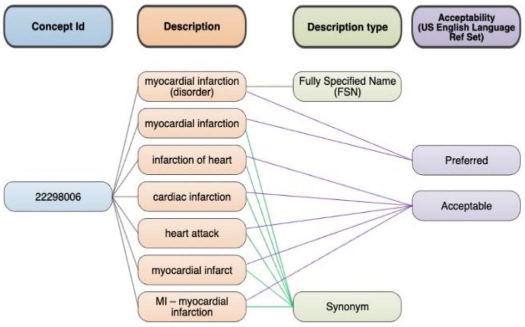


Figure 5. Representation of a collapsed state of *myocardial infarction*

The extended practice of representing terms in a collapsed form may be due to the difficulty of representing more complex systems, but the likely reason for this appears to be that terms had not been seen before as the quantum systems proposed by the QTT. Due to their complexity, the representation of quantum systems is generally mathematical. The mathematical formulation of a quantum system is independent of the type of system; therefore, we can represent a *quanterm* of an undetermined number of states with the equation in Figure 6:

$$|\Psi(t)\rangle = \sum_n C_n(t)|\Phi_n\rangle.$$

Figure 6. Mathematical formulation of a *quanterm*

The psi symbol at the left of the equation is the conventional notation for a system in superposition, that is, a *quanterm* in our case. It is equal to the sum of the amplitude probabilities of observing particular states, where *n* stands for the number of states of the system.

[1] SNOMED CT Starter Guide at https://confluence.ihtsdotools.org/display/DOCSTART

In contrast, the mathematical formulation of collapsed *quanterm* representations, such as the ones in Figures 4 and 5, is simpler:

$$P_j = |a_j\rangle\langle a_j|$$

Figure 7. Mathematical formulation of a collapsed *quanterm*

This equation uses the projection operator P_j to show the projection of the quantum system onto the state $|a_j\rangle$ associated with the measurement outcome a_j (i.e., a definition, conceptualization, feature structure, etc.).

The main challenges related to the QTT are, then, *1)* to represent *quanterms* either by using traditional formats, which seem to be limited for this purpose, or by innovating more sophisticated formats and *2)* to take advantage of such representations to use the potential of *quanterms* for faster and more efficient and intelligent language processing tasks. In quantum mechanics, an electromagnetic wave is sent to the quantum object to verify superposition and to learn about the potential states of the object. According to the equation in Figure 6, the key knowledge learned seems to be the probability of a state happening at a given observation of the object. In times of deep learning and artificial intelligence, language models may play the role of this wave to determine such probabilities.

The representation of a *quanterm* like *mutation* should include, then, the probabilities to predict not only its potential classes but also other variations in its features. These probabilities will make even more sense if they are conditioned by and linked to any relevant entanglement with other *quanterms* and with its context itself.

On the other hand, the potential combination of efficient representation of *quanterms* with modern supercomputing may be necessary. The optimal utilization of *quanterms* and their representation may add to the newly born quantum semantics landscape (see an example of a work that attempts term entanglement in Surov et al., 2021).

5. Conclusions

This paper presents some of the challenges of meaning representation of terms in the light of a quantum theory of terms (QTT) recently reported by Burgos et al. (2024). Due to the novelty of the QTT, we summarized the theory and coined the expression *quanterm* to denote terms viewed as quantum systems. Our focus in this work, however, was on the limitations of representation forms traditionally used in terminology and on the need for innovative representations to respond to the nature of *quanterms*. We highlighted that those traditional representations of terms actually record collapsed *quanterms* at the expense of other potential states (i.e., conceptual variations) of the documented terms.

A comparison of the mathematical formulation of *quanterms* versus collapsed *quanterms* showed the complexity that is being lost in current representations. It was noted that the probabilities to predict particular states of a quantum system are key, not only to this mathematical formulation, but also to potential envisioned forms of *quanterm* representations. Finally, term superposition and entanglement may play an important role not only in term extraction and collocation identification but also in text categorization and knowledge representation, understanding, and generation.

6. Bibliographical References

Adelstein, A. (2007). *Unidad léxica y significado especializado: modelo de representación a partir del nombre relacional madre* [Tesis doctoral, Universidad Pompeu Fabra].

Berri, M. (2013). Léxico generativo y aplicaciones lexicográficas: Los nombres concretos del dominio de la medicina en el DRAE. *Revista signos*, 46(82), 190–212.

Burgos, D. & Vásquez, D. (2024). El nombre terminológico. En G. Quiroz, D. Burgos, & F. Zuluaga (Eds.), *Terminología del español: el término*. Routledge.

Burgos, D. (2024). El término complejo. En G. Quiroz, D. Burgos, & F. Zuluaga (Eds.), *Terminología del español: el término*. Routledge.

Burgos, D., Quiroz, G. & Pérez-Pérez, C.M. (2024). Antecedentes y principios para una teoría cuántica del término. En G. Quiroz, D. Burgos, & F. Zuluaga (Eds.), *Terminología del español: el término*. Routledge.

Cabré, T. (1999). Hacia una aproximación teórica de base comunicativa. Elementos para una teoría de la terminología: hacia un paradigma alternativo. In M. Cabré (Ed.), *La terminología: representación y comunicación. Elementos para una teoría de base comunicativa y otros artículos* (pp. 69–92). Instituto Universitario de Lingüística Aplicada.

Carpenter, B. (1992). The logic of typed feature structures: Applications to unification grammars, logic programs, and constraint resolution. *Cambridge tracts in theoretical computer science*. Cambridge University Press.

Dahlgren, K. (1988). *Naive semantics for natural language understanding*. Kluwer Academic Publishers.

Faber, P., y L'Homme, M. (2022). *Theoretical perspectives on terminology: Explaining terms, concepts, and specialized knowledge*. John Benjamins Publishing Company.

Fellbaum, C. (1998). WordNet: An electronic lexical database. MIT Press.

Francez, N., y Wintner, S. (2011). *Unification grammars*. Cambridge University Press.

Freixa, J. (2005). Variación terminológica: ¿por qué y para qué? *Meta*, 50(4), https://doi.org/10.7202/019917ar

Instituto Colombiano de Normalización (Icontec) e International Organization for Standarization (ISO). (2013). *Trabajo terminológico. Principios y métodos* (NTC-ISO 704).

Mahecha, V., y DeCesaris, J. (2011). Representing Nouns in the Diccionario de aprendizaje del español como lengua extranjera (DAELE). In I. Kosem y K. Kosem (Coord.), *Electronic lexicography in the 21st century: New applications for new users: Proceedings of eLex* (pp. 180–186). Bled.

Miret, S. (2015). Mecánica cuántica. CSIC – Consejo Superior de Investigaciones Científicas.

Pustejovsky, J. (1995). *The generative lexicon*. MIT Press.

Pustejovsky, J. (2011). Coercion in a general theory of argument selection. *Linguistics*, 49(6), 1401–1431.

Surov, I. A., Semenenko, E., Platonov, A. V., Bessmertny, I. A., Galofaro, F., Toffano, Z., ... & Alodjants, A. P. (2021). Quantum semantics of text perception. *Scientific Reports*, 11(1), 4193.

Torres, F. N. (2013). La representación léxica en el modelo del Lexicón Generativo de James Pustejovsky. *Onomázein*, (28), 337-345.

Villanueva, A., López, C., y Ruiz, A. (1999). *Diccionario Mosby: de medicina, enfermería y ciencias de la salud*. Mosby.

VOLARE – Visual Ontological LAnguage REpresentation

Werner Winiwarter
University of Vienna, Faculty of Computer Science
Währingerstrasse 29, 1090 Vienna, Austria
werner.winiwarter@univie.ac.at

Abstract

In this paper, we introduce a novel meaning representation, which is based on AMR but extends it towards a visual ontological representation. We visualize concepts by representative images, and roles by emojis. All concepts are identified either by PropBank rolesets, Wikipedia page titles, WordNet synsets, or Wikidata lexeme senses. We have developed a Web-based annotation environment enabled by augmented browsing and interactive diagramming. As first application, we have implemented a multilingual annotation solution by using English as anchor language and comparing it with French and Japanese language versions. Therefore, we have extended our representation by a translation deviation annotation to document the differences between the language versions. The intended user groups are, besides professional translators and interpreters, students of translation, language, and literary studies. We describe a first use case in which we use novels by French authors and compare them with their English and Japanese translations. The main motivation for choosing Japanese is the soaring popularity of Japanese courses at our university and the particular challenges involved with trying to master this language.

Keywords: meaning representation, AMR, visual annotation, Web-based annotation environment, multilingual annotation, translation annotation, Japanese

1. Introduction

In recent years, there have been many significant developments in the field of designing meaning representations. The most influential approach has been Abstract Meaning Representation (AMR), which again has inspired a wealth of research work to develop parsers and other tools for AMR.

Building on these great efforts, we have extended AMR towards a multilingual representation by adding a translation deviation annotation. Originally, its main intended use has been within the scope of a more far-reaching international research initiative with the aim to assist interpreters and translators with the task of familiarizing themselves with new domain-specific topics (Wloka et al., 2022).

Beyond that we also target educational applications for students of translation, literary, and language studies. In this context we intend to enable classroom scenarios with individual annotation tasks, where the personal knowledge bases can be compared and aggregated for instructional use.

We have implemented a use case of a Web-based annotation environment, which makes it possible to study novels by French authors and compare them with their English and Japanese translations. Japanese was chosen mainly because of the global manga craze, which led to an unprecedented increase in demand for Japanese language courses and, consequently, technological support.

As an important prerequisite for such a scenario we use English as anchor language and map all concepts to disambiguated unique sense identifiers. Each concept is visually represented by an image. For Wikipedia pages, we allocate and download the image through the corresponding Wikidata entry. Other concepts are associated with images from a collection, which we created in our previous research (Winiwarter and Wloka, 2022). This image database contains currently over 3,500 images from Wikimedia Commons of which more than 60% represent abstract concepts. All images are manually selected and annotated with semantic tags and links to WordNet synsets. We also visually represent all roles by using emojis. For that purpose, we map core roles to suitable thematic roles.

For the rendering at the Web client, we use an interactive diagramming library so that the user can freely edit any aspect of the annotation to offer optimal customizability. For example, the user can update the mapping rules from AMR concepts to uniquely identifiable sense definitions, the links between concepts and links, as well as the links between roles and emojis.

After the successful evaluation of our use case implementation in university courses and with professional translators and interpreters, we will make our Web-based annotation environment freely available at GitLab.

This paper is organized as follows. In Sect. 2 we provide related work on topics relevant for this research including some background on Japanese as far as it is helpful for a better understanding; in Sect. 3 we first discuss the design of the meaning representation and user interface; in Sect. 4 we then describe implementation details; and in Sect. 5 we finish with an outlook towards future work.

5th Workshop on Designing Meaning Representations (DMR 2024) @LREC-COLING-2024, pages 54–65
21 May, 2024.

2. Related Work

2.1. Meaning Representations

The annotation of sentences with *meaning representations* has established itself in the last decade as a thriving research field in computational linguistics (see Abend and Rappoport, 2017). The most influential and most actively promoted approach has been the *Abstract Meaning Representation*[1] (AMR) (Banarescu et al., 2013). There are many parsers available, the best[2] parser being at the moment Lee et al. (2022). The *SPRING* parser (Bevilacqua et al., 2021) can be tried via a Web interface[3], which also offers a nice visualization. One point of criticism concerning AMR's reliance on numbered, not directly interpretable core arguments, is addressed by the *WISeR* meaning representation (Feng et al., 2023), which maps them to thematic roles. There also exist several AMR annotation tools, one recent Web-based solution is *CAMRA* (Cai et al., 2023).

AMR has been recently extended to the *Uniform Meaning Representation*[4] (UMR) (Gysel et al., 2021). It enhances AMR by adding support for other languages (in particular low-resource languages), and a document-level representation capturing intersentential coreference and temporal/modal dependencies. There is an upcoming workshop to kick-start the development of UMR parsers[5].

According to the *UMR guidelines*[6], UMR fully embraces *radical construction grammar* as a theoretical foundation (Croft, 2001, 2022), which was designed with *typological* (Croft, 2002) applicability as main motivation, i.e. to study and classify languages according to their structural features to allow their comparison. Radical construction grammar considers word classes and other syntactic structures as language-specific and construction-specific (Croft, 2023).

2.2. Multimodality

Multimodal enhancements of lexical resources have a long history but only recently gained new momentum due to the strong interest in research on *visual question answering* (VQA) (Lerner et al., 2024) or *multimodal large language models* (MLLMs) (Bewersdorff et al., 2024). One example of an attempt towards a multimodal semantic representation is *VoxML* (Pustejovsky et al., 2016).

Regarding the mapping of images to WordNet synsets, there exists the *ImageNet* collection, which maps ca. 1,000 images to each synset (Deng et al., 2009). Another effort to assign cliparts to synsets was discontinued after illustrating only 581 synsets (Bond et al., 2009). A much more influential resource is *Wikipedia*, which has been increasingly enhanced with visual representations. However, the number of images varies widely across language versions. The most comprehensive recent effort is certainly *BabelNet*[7] (Navigli et al., 2021) with the annotation tool *Babelfy*[8] (Moro et al., 2014) and the latest *BabelPic*[9] (Calabrese et al., 2020) dataset targeting non-concrete concepts.

There also exists a subfield of *cognitive linguistics* dealing with identifying and analyzing language-image relations in multimodal texts, e.g. research on intersemiotic convergence (Hart and Queralto, 2021). One central term in this context is *grounding*, which is interpreted in quite different ways by natural language processing and cognitive science researchers (see Chandu et al., 2021). Whereas natural language processing emphasizes the linking of text to other modalities, cognitive science focuses on how speakers build the common ground to share mutual information. During this cognitive process, a set of abstract symbols acquire meaning through perceptions and situated actions (Chen et al., 2023) in analogy to the concept of *construal* in cognitive linguistics (Langacker, 2008), which accounts for choosing alternative linguistic expressions for expressing the same situation (Divjak et al., 2020).

The use of pictorial illustrations has a long history in language teaching didactics and there exist numerous empirical studies that show their effectiveness at all levels of proficiency, e.g. Tahiri (2020). Nonetheless, to the best of our knowledge, we are not aware of any related work with the aim of creating visual representations of meaning representations of sentences.

2.3. Translation Deviations

While there is an ample supply of tools for translators (Rothwell et al., 2023), there have been comparatively few research efforts on annotating translation deviations. One example is Deng and Xue (2017), who analyzed deviations between Chinese and English texts produced by machine translation. There exists a related research work on creating corpora of machine translated documents with annotated translation errors (Fishel et al., 2012),

[1] `https://amr.isi.edu/`

[2] `https://paperswithcode.com/task/amr-parsing/latest`

[3] `http://nlp.uniroma1.it/spring/`

[4] `https://umr4nlp.github.io/web/`

[5] `https://umr4nlp.github.io/web/UMRParsingWorkshop.html`

[6] `https://github.com/umr4nlp/umr-guidelines/`

[7] `https://babelnet.org/`

[8] `http://babelfy.org/`

[9] `https://sapienzanlp.github.io/babelpic/`

and a more recent work on creating an English-French-Chinese corpus annotated with translation relations (Zhai et al., 2018).

2.4. Japanese Language

Japanese is an agglutinative SOV language with topic-comment sentence structure. Phrases are exclusively head-final, and compound sentences are strictly left-branching. The most noticeable characteristics for language students are the missing articles, no distinction between singular and plural, no gender, no conjugation for person, a complex system of honorifics, and a high level of ambiguity, e.g. by omitting the subject or using zero anaphora. There exist many excellent reference grammars, e.g. Kamermans (2010); Kaiser et al. (2013), and a lot of research activity on Japanese linguistics (see Hasegawa, 2015, 2018).

One of the main obstacles for getting proficient in Japanese is the complex writing system (see Matsumoto, 2007; Mori, 2014; Paxton, 2019). It uses a combination of logographic *kanji* and two syllabaries *hiragana* and *katakana*. Kanji are adopted Chinese characters, since 2010 Japanese students are required to learn 2,136 so-called *jōyō kanji* in primary and secondary school. Most kanji have more than one reading depending on the context.

The most important lexical resource for Japanese is the *Japanese Multilingual dictionary* (JMdict) (Breen, 2004), which can be searched online in combination with many other lexical resources via the *Online Japanese Dictionary Service* (WWWJDIC)[10].

Another very useful online service is *Honyaku Star*[11]. It references numerous dictionaries and corpora and shows translations in context. Honyaku Star includes currently over 2 million translations. Japanese is also part of the *Open Multilingual Wordnet* (OMW) (Bond and Paik, 2012)[12], which makes it possible to assign Japanese words to English synsets. OMW is easily accessible via the *NLTK* toolkit[13].

The most prolific linguistic tool for Japanese is certainly the *CaboCha* dependency parser (Kudo and Matsumoto, 2002), which includes the *MeCab* part-of-speech and morphological analyzer (Kudo et al., 2004). More recently, trained pipelines have been added to the popular natural language toolkit *SpaCy*[14], another similar solution is *UniDic2UD*[15].

3. User Interface

In this section, we introduce our meaning representation by providing examples of the visual rendering in the user interface. The technical details are addressed later in Sect. 4.

The choices leading to the current user interface design are mainly based on practical experience and user feedback from previous research on meaning representation (Wloka and Winiwarter, 2021a), kanji acquisition (Wloka and Winiwarter, 2021b), and multimodal analogies (Winiwarter and Wloka, 2023) as part of a Web-based Japanese language learning environment. In all three cited publications we exclusively used images from Wikimedia Commons which are embedded in Wikipedia pages. The decision to restrict ourselves to this image source was mainly motivated by licensing issues but also by the valuable contextual semantic information accessible through the links to the original Wikipedia page(s).

The idea of using emojis to represent roles originated from our previous research work on using kanji within educational strategic games to foster incidental learning (Winiwarter, 2017). In recent implementations we increasingly relied on emojis to communicate additional gameplay information. This way we successfully assisted the user in focusing on kanji by eliminating other textual elements from the display. For a recent survey of research on emojis we refer to Bai et al. (2019).

As running example text for showcasing our user interface design, we use “From the Earth to the Moon”[16], an 1865 novel by Jules Verne, together with its English[17] and Japanese[18] translations. Figure 1 shows the AMR of the first sentence of the English version displayed in the *AMR Editor*[19]. The corresponding VOLARE representation is shown in Fig. 2. Each concept is visualized by a representative image. Whenever the user hovers over a concept, a tooltip with the concept identifier is displayed. For the ease of the reader, we have added the tooltip texts to Fig. 2 and the following user interface figures. A click on a concept shows an enlarged version of the image as well as the concept identifier and gloss in the bottom right corner of the screen.

In addition to *PropBank rolesets*, color-coded in teal, we can also observe *Wikipedia page titles* as concept identifiers shown in magenta:

- club → `Club_(organization)`,

[10] http://wwwjdic.se/

[11] http://honyakustar.com/

[12] https://omwn.org/

[13] https://www.nltk.org/

[14] https://spacy.io/models/ja

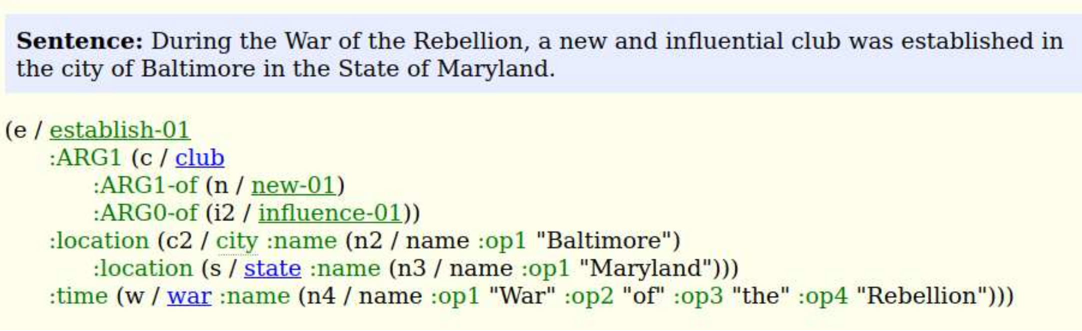
[15] https://github.com/KoichiYasuoka/UniDic2UD

[16] https://fr.wikisource.org/wiki/De_la_Terre_à_la_Lune

[17] https://en.wikisource.org/wiki/From_the_Earth_to_the_Moon

[18] https://ja.wikisource.org/wiki/地球から月へ

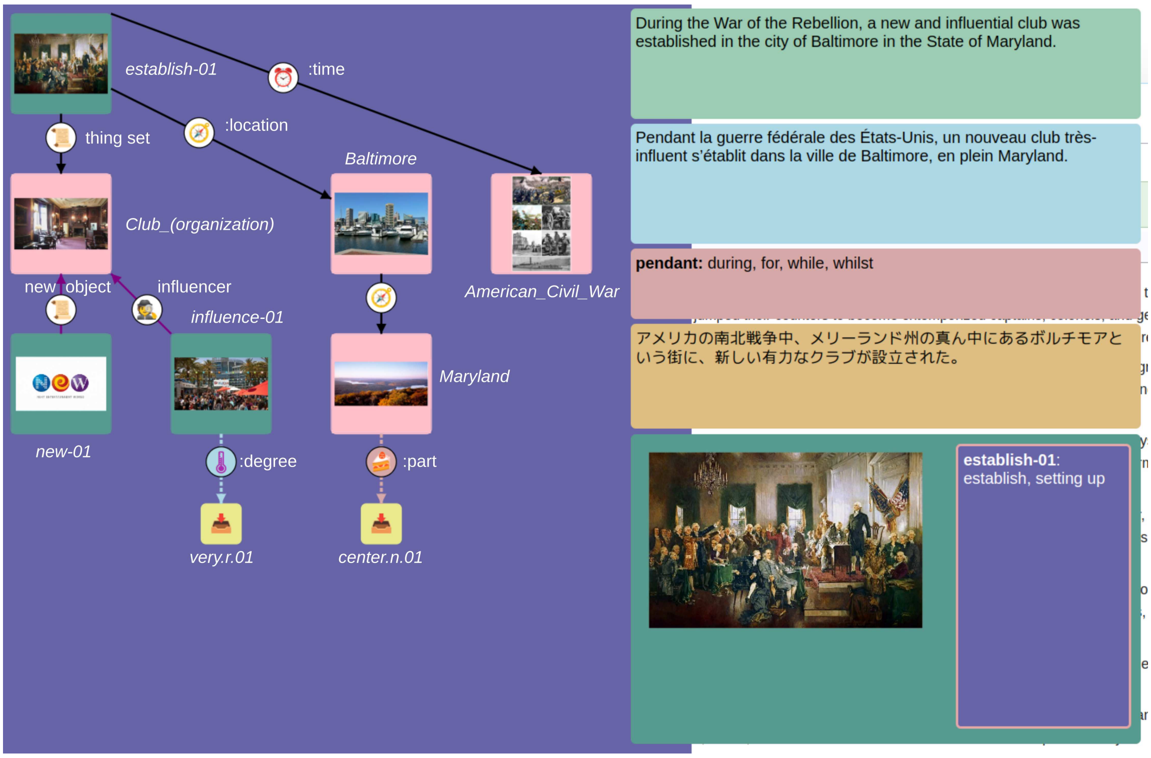
[19] https://amr.isi.edu/editor.html



Sentence: During the War of the Rebellion, a new and influential club was established in the city of Baltimore in the State of Maryland.

```
(e / establish-01
    :ARG1 (c / club
        :ARG1-of (n / new-01)
        :ARG0-of (i2 / influence-01))
    :location (c2 / city :name (n2 / name :op1 "Baltimore")
        :location (s / state :name (n3 / name :op1 "Maryland")))
    :time (w / war :name (n4 / name :op1 "War" :op2 "of" :op3 "the" :op4 "Rebellion")))
```

Figure 1: Example AMR.



During the War of the Rebellion, a new and influential club was established in the city of Baltimore in the State of Maryland.

Pendant la guerre fédérale des États-Unis, un nouveau club très-influent s'établit dans la ville de Baltimore, en plein Maryland.

pendant: during, for, while, whilst

アメリカの南北戦争中、メリーランド州の真ん中にあるボルチモアという街に、新しい有力なクラブが設立された。

establish-01: establish, setting up

Figure 2: Example of user interface.

- Baltimore → Baltimore,
- Maryland → Maryland,
- War of the Rebellion → American_Civil_War.

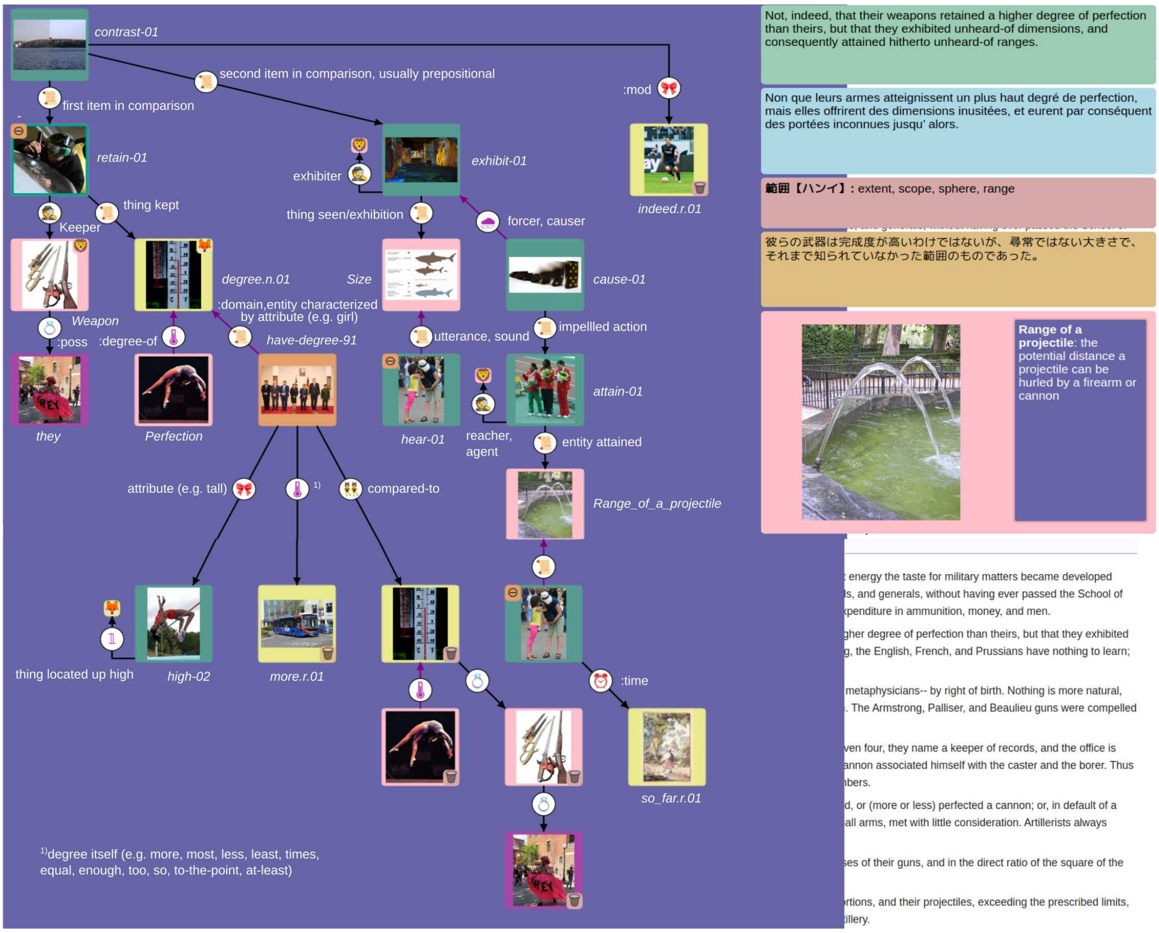
As can be seen, this leads to a more concise and uncluttered representation of named entities.

AMR roles are visualized using *emojis*, the original roles are available as tooltips. For core arguments, we choose the emoji according to the role indicated in the PropBank frame file and show the gloss as tooltip. Inverse roles are indicated by inverted arrows and the line color violet. On the right side in Fig. 2 we display the sentence in English, French, and Japanese. By clicking on a word in either the French or Japanese version, lexical information with English glosses can be displayed, e.g. for the French word "pendant" in this case.

One important extension of AMR are the two *translation deviation annotations* (TDAs) in Fig. 2. The left one represents the word "très" in the French original, which is missing in the English translation. It is mapped to the *WordNet synset* very.r.01 and therefore color-coded in yellow.

To keep the representation manageable, we only show an emoji, however, the detailed information can still be displayed in the bottom right corner. The relation for the role :degree (🌡️) is drawn as dashed line in the color cyan to indicate the language French.

In the same way, we map the two additional expressions "en plein" and "真ん中" to the synset center.n.01 and link it to Maryland by the role :part (🍰). This line is drawn in purple to indicate



Not, indeed, that their weapons retained a higher degree of perfection than theirs, but that they exhibited unheard-of dimensions, and consequently attained hitherto unheard-of ranges.

Non que leurs armes atteignissent un plus haut degré de perfection, mais elles offrirent des dimensions inusitées, et eurent par conséquent des portées inconnues jusqu' alors.

範囲【ハンイ】: extent, scope, sphere, range

彼らの武器は完成度が高いわけではないが、尋常ではない大きさで、それまで知られていなかった範囲のものであった。

Range of a projectile: the potential distance a projectile can be hurled by a firearm or cannon

energy the taste for military matters became developed
ls, and generals, without having ever passed the School of
xpenditure in ammunition, money, and men.

gher degree of perfection than theirs, but that they exhibited
g, the English, French, and Prussians have nothing to learn;

metaphysicians-- by right of birth. Nothing is more natural,
. The Armstrong, Palliser, and Beaulieu guns were compelled

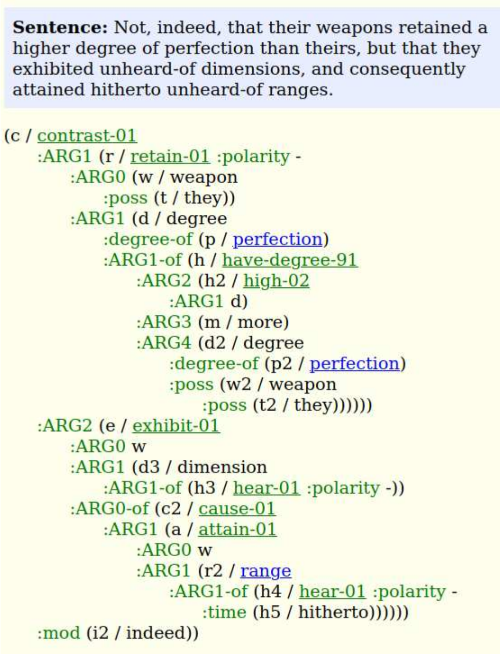
ven four, they name a keeper of records, and the office is
annon associated himself with the caster and the borer. Thus
nbers.

d, or (more or less) perfected a cannon; or, in default of a
all arms, met with little consideration. Artillerists always

ses of their guns, and in the direct ratio of the square of the

rtions, and their projectiles, exceeding the prescribed limits,
illery.

Figure 3: Example of user interface with co-reference.



Sentence: Not, indeed, that their weapons retained a higher degree of perfection than theirs, but that they exhibited unheard-of dimensions, and consequently attained hitherto unheard-of ranges.

```
(c / contrast-01
    :ARG1 (r / retain-01 :polarity -
        :ARG0 (w / weapon
            :poss (t / they))
        :ARG1 (d / degree
            :degree-of (p / perfection)
            :ARG1-of (h / have-degree-91
                :ARG2 (h2 / high-02
                    :ARG1 d)
                :ARG3 (m / more)
                :ARG4 (d2 / degree
                    :degree-of (p2 / perfection)
                    :poss (w2 / weapon
                        :poss (t2 / they))))))
    :ARG2 (e / exhibit-01
        :ARG0 w
        :ARG1 (d3 / dimension
            :ARG1-of (h3 / hear-01 :polarity -))
        :ARG0-of (c2 / cause-01
            :ARG1 (a / attain-01
                :ARG0 w
                :ARG1 (r2 / range
                    :ARG1-of (h4 / hear-01 :polarity -
                        :time (h5 / hitherto))))))
    :mod (i2 / indeed))
```

Figure 4: Example AMR with co-reference.

both French and Japanese (just Japanese would be orange).

In VOLARE, we only use variables for *co-reference*. Figure 3 and Fig. 4 give an example. We use animal emojis as variables (e.g. 🦁, 🦊). To avoid overloading the display, we draw the variable at the top right corner of the concept and duplicate it as target of co-referential links.

As can be seen, we visualize *negation* of concepts with a ⊖ symbol in the top left corner. Additional concept types used in this example are *Wikidata lexeme senses* (violet) and *special AMR frames* (orange).

Finally, the *wastebaskets* (🗑️) in the bottom right corners are translation deviation annotations that indicate concepts that are missing in the respective languages. The lexical information shown in Fig. 3 is for the Japanese word "範囲". It provides the pronunciation【ハンイ】in katakana, which corresponds to "han'i", and the English glosses "extent, scope, sphere, range".

One common case of translation deviation is the *substitution* of one or several concepts in the source language by alternative concepts in the target language. An example of a simple concept substitution is shown in Fig. 5. The concept displayed in the image for the English text is `Pocket_pistol`, which is an exact translation of the French original “pistolets de poche”. However, in Japanese it is translated as 拳銃, which can be mapped to the concept `Pistol`. This deviation is indicated by the symbol ⇄ in the bottom right corner. By clicking on the symbol, the user can inspect the detailed information about this concept.

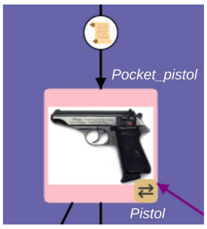


Figure 5: Example of simple substitution.

A more complex situation is depicted in Fig. 6. The corresponding AMR snippet is:

```
(b / become-01
 :ARG1 (t / taste
   :topic (m / matter
     :topic (m2 / military)))
  ...
```

In this case, the original French expression for “the taste for military matters” is “ l’instinct militaire”. This is indicated by a substitute concept `Instinct` with the relation `:ARG1` (`entity changing`) from `become-01` and a relation `:mod` to `Military`. Since the Japanese text offers a precise translation of the French original, the deviation from the English version concerns both languages.

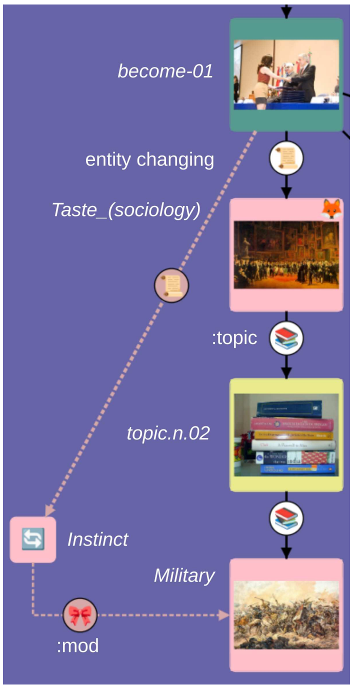


Figure 6: Example of complex substitution.

4. Implementation

In this section we will describe some details about the implementation of our Web-based annotation environment. We first provide a top-level overview of the system architecture, before we zoom in on the three subtasks `Preprocess`, `Annotate`, and `Customize` in separate subsections.

Figure 7 highlights the main components of our architecture. The users can access the server through a Web browser by using augmented browsing enabled through *Chrome extension APIs*[20], and the *jQuery*[21] and *jQuery UI*[22] libraries.

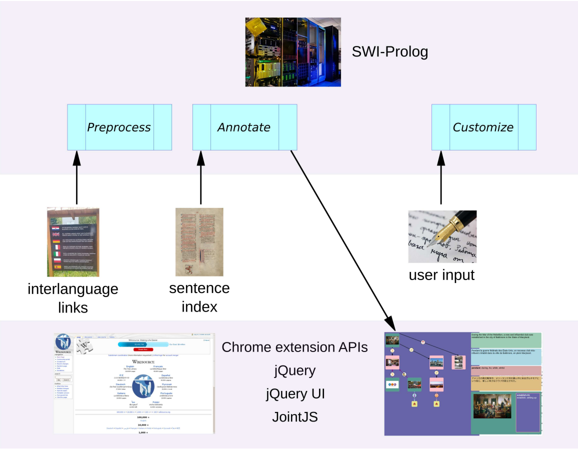


Figure 7: System architecture.

If a student loads a new Wikisource document in one of the three languages English, French, or Japanese, it is automatically analyzed and segmented into individual sentences. Each sentence is augmented with an event handler so that whenever a student then clicks on a sentence, it is transferred to the server.

If the Web document is a new text, we use the *interlanguage links* to retrieve the other two language versions and `Preprocess` the resulting document triplet. For existing texts, we use the *sentence index* to identify the selected sentence and `Annotate` it to produce the VOLARE annotation, which is sent back to the user. The rendering at the Web client is realized using the JavaScript interactive diagramming library *JointJS*[23] based on SVG. Among the many available diagramming solutions, we chose JointJS mainly because it is open

[20] https://developer.chrome.com/docs/extensions/reference/api

[21] https://jquery.com/

[22] https://jqueryui.com/

[23] https://www.jointjs.com/

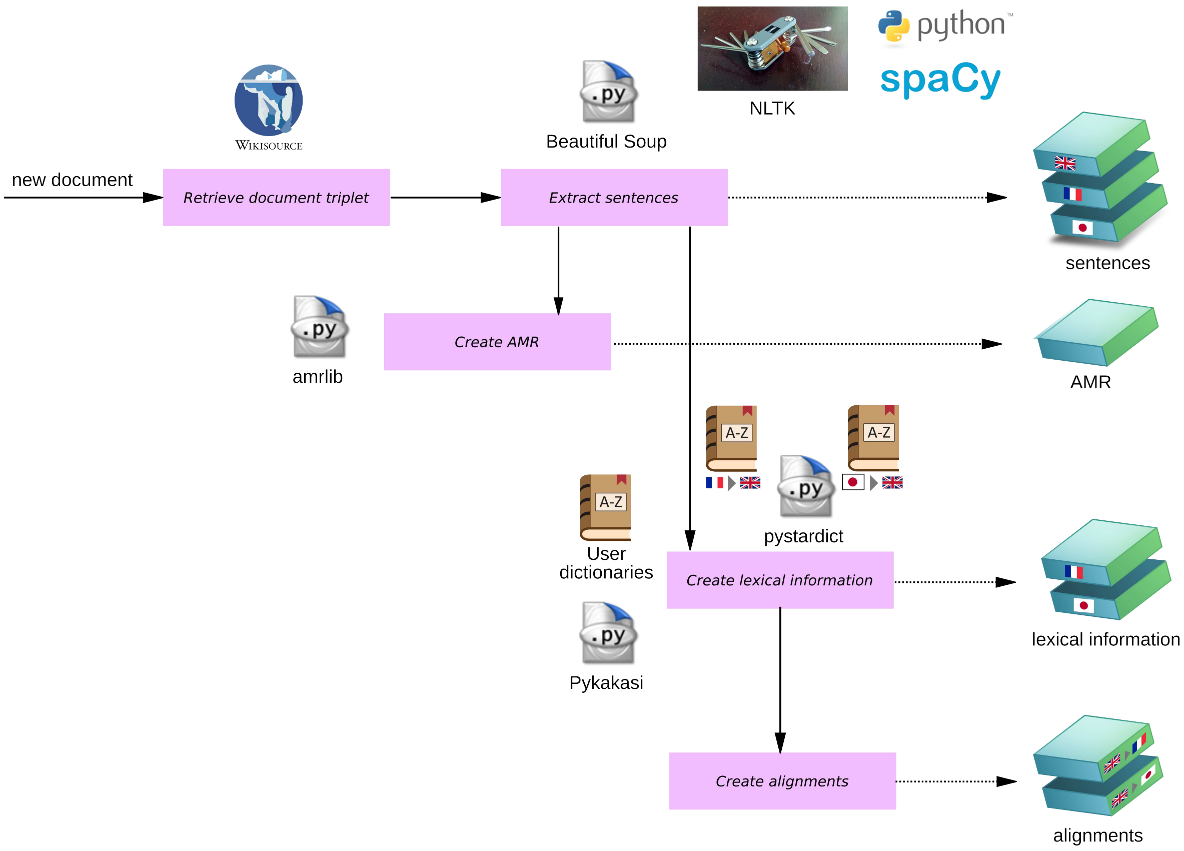


Figure 8: Subtask Preprocess.

source software[24], feature-rich, compatible with our augmented browsing scenario, and offers excellent documentation with many tutorials and demos.

The interactive diagramming library makes it possible to freely edit any aspect of the presented annotation. Each *user input* is sent to the server and leads to an update of the personal knowledge base used to `Customize` the annotation.

The annotation server is implemented in *SWI-Prolog*[25] (Wielemaker et al., 2012), which is not only an obvious choice for natural language processing tasks but also provides a scalable Web server solution (Wielemaker et al., 2008) and libraries for efficiently handling RDF and XML files.

4.1. Subtask Preprocess

Figure 8 gives an overview of the individual steps to preprocess a *new document*. In general, we have mainly written *Python* scripts for that purpose. In addition, we make use of the popular NLP toolkits *NLTK* and *SpaCy*. The latter provides trained models and pipelines for several languages including English, French, and Japanese.

The first step for a new English, French, or Japanese document is to `Retrieve the document triplet` by downloading the other two language versions from *Wikisource*. The three documents are then parsed by using the Python library *Beautiful Soup*[26] to `Extract the individual sentences`. Based on the resulting sentence collection, we can start to `Create` the `AMR` and the `lexical information`. These two steps can therefore be performed in parallel. We use the Python library *amrlib*[27] to produce the AMR, which is available as SpaCy extension[28].

To create the lexical information, in particular English glosses for French and Japanese words and expressions, we use the Python library *pystardict*[29] to access the French-English and the Japanese-English *StarDict dictionaries*[30]. The latter is based on the popular JMdict dictionary. In addition, we look up the personal *user dictionaries* (see Sect. 4.3) to include customized entries.

[24] https://sourceforge.net/projects/jointjs/
[25] https://www.swi-prolog.org
[26] https://www.crummy.com/software/BeautifulSoup/
[27] https://github.com/bjascob/amrlib
[28] https://spacy.io/universe/project/amrlib
[29] https://github.com/lig/pystardict
[30] https://stardict-4.sourceforge.net/

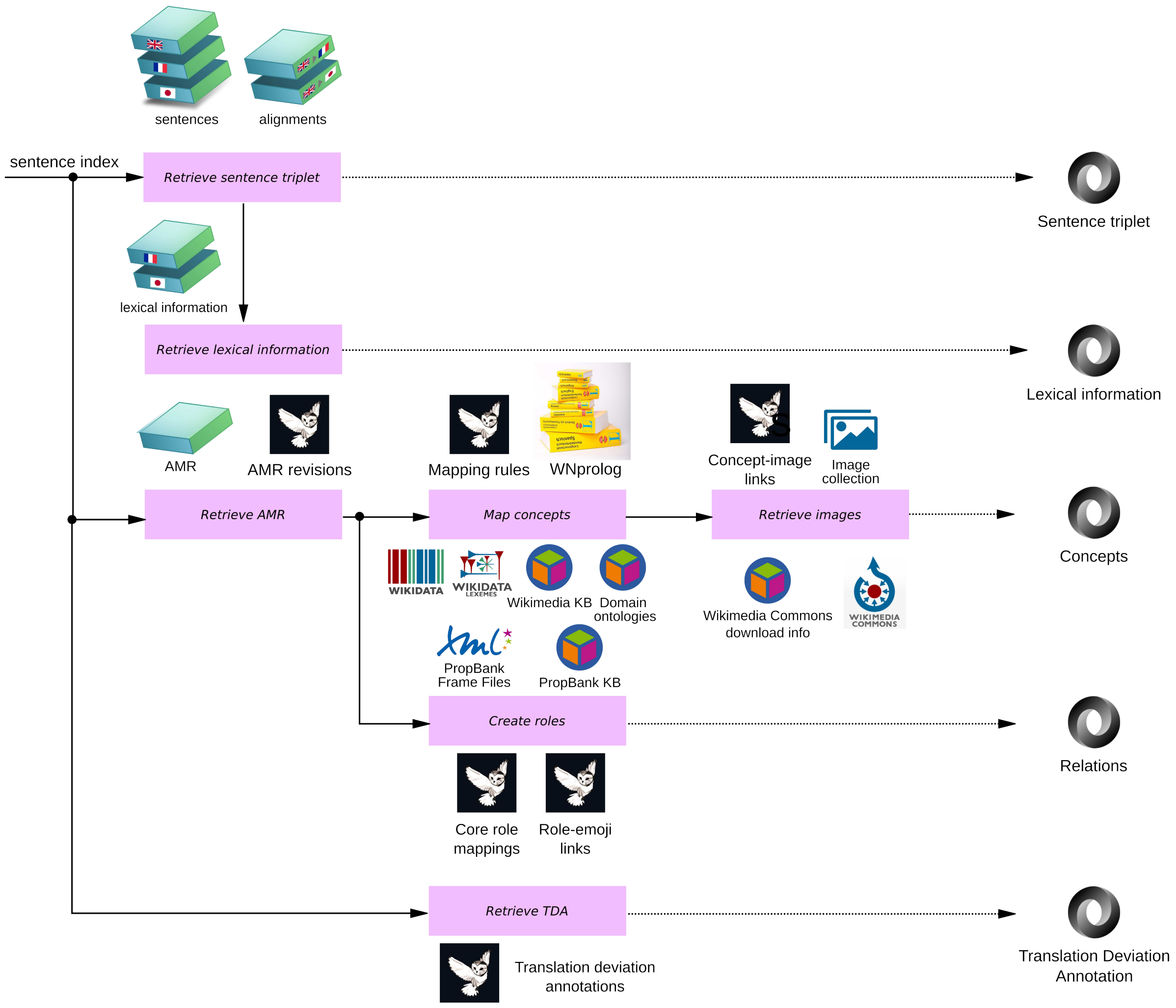


Figure 9: Subtask Annotate.

Finally, we use the Python library *Pykakasi*[31] based on the transliteration tool *kakasi*[32] to add the correct pronunciation to Japanese words with kanji characters. The last preprocessing step is to `Create` the bilingual `alignments` between the language pairs English-French and English-Japanese. Based on the lexical information we calculate a similarity value to align sentences depending on the comparison with threshold values. Although $1 : 1$ and $1 : 2$ are the most common cases, we are able to handle all theoretically possible $0 \ldots n : 0 \ldots m$ patterns.

For performance reasons, we first execute the necessary preprossing steps for the sentence that the user wants to inspect so that the annotation subtask can start without any significant delay for the user. This means that the remainder of the document is analyzed as background process while the user can already interact with the annotation at the client. We also keep all the required SpaCy models and pipelines preloaded to save valuable initialization time.

4.2. Subtask Annotate

The details of this subtask are depicted in Fig. 9. Based on the resources created in the previous subsection, we generate *JSON* objects for the individual components of the annotation. We apply various types of customization data provided by the user (see Sect. 4.3) in several processing steps.

To produce the list of concepts for rendering in VOLARE, we first `Retrieve` the `AMR` and correct it according to the *AMR revisions*. The next step is to `Map` the `concepts` in the AMR to uniquely identifiable sense definitions. For *PropBank rolesets*, we only have to add glosses from our *PropBank KB*, which we extracted from the *PropBank Frame Files*, which are in *XML* format. All other concepts, we first try to match with *Wikipedia pages* and the corresponding *Wikidata items*. The necessary information is stored in our *Wikimedia KB*, which we generated with the use of *DBpedia* (Lehmann et al.,

[31] https://pypi.org/project/pykakasi/
[32] http://kakasi.namazu.org/

2015) datasets. Next, we attempt to find compatible *WordNet* (Princeton University, 2012) synsets via *WNprolog*. Since existing word sense disambiguation solutions such as *pywsd*[33] produced unsatisfactory results, we have developed our own model, which we continuously improve based on user input stored as *mapping rules*. Any remaining concepts are mapped to *Wikidata Lexemes*. Finally, the user can add *domain ontologies* to this collection of ontological resources.

The last missing task for creating the concept description is to `Retrieve` the representative `images`. We store for all Wikipedia pages in our Wikimedia KB the *Wikimedia Commons download info* so that we can download any images for concepts that are accessed for the first time. WordNet synsets are mapped to images in our *image collection* according to the available synset associations. The users can add their own *concept-image links* to personalize their visualization.

For the relation description, one important task is to `Create` the `role` for display by mapping core roles to thematic roles according to the defined *core role mappings*, and to translate the role names to emojis by following the *role-emoji links*. Both can be freely adjusted to suit the preferences of the user. The role glosses for the tooltips are retrieved from the PropBank KB. Finally, if there exists any *translation deviation annotation* for this sentence, we `Retrieve` the `TDA` and add it to the JSON array to complete the VOLARE annotation.

4.3. Subtask Customize

All the processing steps in Fig. 8 and Fig. 9 are carried out fully automatically. However, in many cases there is still room for improvement or the desire for adjustments to better adapt the annotation results to individual needs and preferences.

Since we have already covered many aspects of customization in the previous subsections, we can keep the presentation here brief and just summarize the numerous possibilities offered to the user. Enabled by the interactive diagramming functionality of JointJS, the users can freely edit any VOLARE annotation to fine-tune it to better suit their personal preferences. Any user input is sent to the server where it is processed and leads to an update of the affected resources.

Any changes to the original AMR are stored in the *AMR revisions* and used to correct the AMR if the sentence annotation is displayed again in future. In the same way, any *translation deviation annotation* is saved and can be reviewed at a later time. The users can also change concepts in the annotation leading to an update of the *mapping rules*; they can add *domain ontologies*; and add new images to the *image collection* or use existing images to represent a concept, which changes the *concept-image link*. Similarly, they can choose different roles for core arguments, which updates the *core role mappings*, as well as other emojis, which results in an actualized *role-emoji link*. Finally, the users can also improve the display of the lexical information by adding new entries to their *user dictionaries*.

An important aspect is that in classroom or company environments, the individual customization data can be collected, analyzed, and consolidated to create integrated resources at the organizational unit level.

5. Conclusion

In this paper, we have presented a Web-based annotation environment, which extends AMR with visual and ontological elements. The addition of TDAs enables the comparative analysis of the different language versions of a document.

We will evaluate our use case implementation with several volunteer professional translators and interpreters. Based on the achieved results including user feedback concerning functionality and usability aspects, we will further improve our system and draw up annotation guidelines for the users.

Interesting aspects for the evaluation will be the impact of adding representative images on the quality of the translation process, a detailed analysis of the customizations performed by the users, and experiments from a cognitive linguistic perspective, e.g. regarding the question whether a multilingual speaker has a single cross-lingual visual representation of a concept or different visual representations depending on the language currently used.

We have also already planned classroom scenarios in university courses to investigate the educational benefits of our environment and the challenges/opportunities that arise from aggregating and harmonizing the individual customization data.

The extension to other languages is straightforward as long as StarDict dictionaries and SpaCy models and pipelines exist. We can also easily accommodate more than three languages, however, this will require some filtering regarding the TDAs, otherwise the display will become too cluttered.

The main challenge for future work is to switch the foundation of VOLARE to UMR, which is also much better suited for multilingual annotations. The incorporation of the UMR document-level representation will make it possible to model intersentential dependencies, which will at the same time lead to new requirements for our environment and TDAs. We have already started with some first considerations and preparatory work. Thus, we eagerly await the availability of UMR parsers to begin with real experiments and implementation work.

[33] https://github.com/alvations/pywsd

6. Bibliographical References

Omri Abend and Ari Rappoport. 2017. The state of the art in semantic representation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 77–89, Vancouver, Canada. Association for Computational Linguistics.

Qiyu Bai et al. 2019. A systematic review of emoji: Current research and future perspectives. *Frontiers in Psychology*, 10.

Laura Banarescu et al. 2013. Abstract Meaning Representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria. Association for Computational Linguistics.

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 (AAAI 2021)*, pages 12564–12573.

Arne Bewersdorff et al. 2024. Taking the next step with generative artificial intelligence: The transformative role of multimodal large language models in science education. arXiv:2401.00832 [cs.AI].

Francis Bond and Kyonghee Paik. 2012. A survey of wordnets and their licenses. In *Proceedings of the 6th Global WordNet Conference (GWC 2012)*, pages 64–71.

Francis Bond et al. 2009. Enhancing the Japanese WordNet. In *Proceedings of the 7th Workshop on Asian Language Resources (ALR7)*, pages 1–8, Suntec, Singapore. Association for Computational Linguistics.

James Breen. 2004. JMdict: A Japanese-Multilingual dictionary. In *Proceedings of the Workshop on Multilingual Linguistic Resources*, pages 71–79. Association for Computational Linguistics.

Jon Z. Cai et al. 2023. CAMRA: Copilot for AMR annotation. arXiv:2311.10928 [cs.CL].

Agostina Calabrese, Michele Bevilacqua, and Roberto Navigli. 2020. Fatality killed the cat or: BabelPic, a multimodal dataset for non-concrete concepts. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4680–4686, Online. Association for Computational Linguistics.

Khyathi Raghavi Chandu, Yonatan Bisk, and Alan W Black. 2021. Grounding 'grounding' in NLP. arXiv 2106.02192 [cs.CL].

Pin-Er Chen et al. 2023. Exploring affordance and situated meaning in image captions: A multimodal analysis. In *Proceedings of the 37th Pacific Asia Conference on Language, Information and Computation*, pages 293–302, Hong Kong, China. Association for Computational Linguistics.

William Croft. 2001. *Radical Construction Grammar: Syntactic Theory in Typological Perspective*. Oxford University Press.

William Croft. 2002. *Typology and Universals*, 2nd edition. Cambridge Textbooks in Linguistics. Cambridge University Press.

William Croft. 2022. *Morphosyntax: Constructions of the World's Languages*. Cambridge Textbooks in Linguistics. Cambridge University Press.

William Croft. 2023. Word classes in Radical Construction Grammar. In *The Oxford Handbook of Word Classes*. Oxford University Press.

Dun Deng and Nianwen Xue. 2017. Translation divergences in Chinese-English machine translation: An empirical investigation. *Computational Linguistics*, 43(3):521–565.

Jia Deng et al. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Miami, FL.

Dagmar Divjak, Petar Milin, and Srdan Medimorec. 2020. Construal in language: A visual-world approach to the effects of linguistic alternations on event perception and conception. *Cognitive Linguistics*, 31(1):37–72.

Lydia Feng et al. 2023. Widely interpretable semantic representation: Frameless meaning representation for broader applicability. arXiv:2309.06460 [cs.CL].

Mark Fishel, Ondřej Bojar, and Maja Popović. 2012. Terra: a collection of translation error-annotated corpora. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 7–14, Istanbul, Turkey. European Language Resources Association (ELRA).

Jens E. L. Van Gysel et al. 2021. Designing a Uniform Meaning Representation for natural language processing. *KI – Künstliche Intelligenz*, 35:343–360.

Christopher Hart and Javier Marmol Queralto. 2021. What can cognitive linguistics tell us about language-image relations? A multidimensional approach to intersemiotic convergence in multimodal texts. *Cognitive Linguistics*, 32(4):529–562.

Yoko Hasegawa. 2015. *Japanese: A Linguistic Introduction*. Cambridge University Press, Cambridge, UK.

Yoko Hasegawa, editor. 2018. *The Cambridge Handbook of Japanese Linguistics*. Cambridge Handbooks in Language and Linguistics. Cambridge University Press.

Stefan Kaiser et al. 2013. *Japanese: A Comprehensive Grammar*, 2nd edition. Routledge, London and New York.

Michiel Kamermans. 2010. *An Introduction to Japanese – Syntax, Grammar & Language*. SJGR Publishing, Rotterdam, The Netherlands.

Taku Kudo and Yuji Matsumoto. 2002. Japanese dependency analysis using cascaded chunking. In *CoNLL 2002: Proceedings of the 6th Conference on Natural Language Learning 2002 (COLING 2002 Post-Conference Workshops)*, pages 63–69.

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, Barcelona, Spain. Association for Computational Linguistics.

Ronald W. Langacker. 2008. *Cognitive Grammar: A Basic Introduction*. Oxford University Press.

Young-Suk Lee et al. 2022. Maximum Bayes Smatch ensemble distillation for AMR parsing. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5379–5392, Seattle, United States. Association for Computational Linguistics.

Jens Lehmann et al. 2015. DBpedia – a large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web*, 6(2):167–195.

Paul Lerner, Olivier Ferret, and Camille Guinaudeau. 2024. Cross-modal retrieval for knowledge-based visual question answering. arXiv:2401.05736 [cs.CL].

Hiroshi Matsumoto. 2007. Peak learning experiences and language learning: A study of American learners of Japanese. *Language Culture and Curriculum - LANG CULT CURRIC*, 20:195–208.

Yoshiko Mori. 2014. Review of recent research on kanji processing, learning, and instruction. *Japanese Language and Literature*, 48(2):403–439.

Andrea Moro, Alessandro Raganato, and Roberto Navigli. 2014. Entity Linking meets Word Sense Disambiguation: a Unified Approach. *Transactions of the Association for Computational Linguistics (TACL)*, 2:231–244.

Roberto Navigli et al. 2021. Ten years of BabelNet: A survey. In *Proceedings of IJCAI 2021*, pages 4559–4567.

Simon Paxton. 2019. Kanji matters in a multilingual Japan. *The Journal of Rikkyo University Language Center*, 42:29–41.

James Pustejovsky et al. 2016. The development of multimodal lexical resources. In *Proceedings of the Workshop on Grammar and Lexicon: interactions and interfaces (GramLex)*, pages 41–47, Osaka, Japan. The COLING 2016 Organizing Committee.

Andrew Rothwell et al. 2023. *Translation Tools and Technologies*, 1st edition. Routledge, London and New York.

Shejla Tahiri. 2020. The impact of pictures on second language acquisition. *SEEU Review*, 15(2):126–135.

Jan Wielemaker, Zhisheng Huang, and Lourens Van Der Meij. 2008. SWI-Prolog and the Web. *Theory and Practice of Logic Programming*, 8(3):363–392.

Jan Wielemaker et al. 2012. SWI-Prolog. *Theory and Practice of Logic Programming*, 12(1-2):67–96.

Werner Winiwarter. 2017. KANGAROO – the kanji game room. In *Proceedings of the 19th International Conference on Information Integration and Web-based Applications & Services*, pages 535–542, New York. ACM.

Werner Winiwarter and Bartholomäus Wloka. 2022. VISCOSE – a kanji dictionary enriched with VISual, COmpositional, and SEmantic information. In *7th Workshop on Cognitive Aspects of the Lexicon (CogALex-VII)*, pages 68–77.

Werner Winiwarter and Bartholomäus Wloka. 2023. CLE-UMA – a creative learning environment using multimodal analogies. In *The 17th International Conference on Knowledge, Information and Creativity Support Systems*, IIAI Letters on Informatics and Interdisciplinary Research.

Bartholomäus Wloka, Yves Lepage, and Werner Winiwarter. 2022. WAPITI – web-based assignment preparation and instruction tool for interpreters. In *24th International Conference on Information Integration and Web Intelligence (iiWAS2022)*, pages 295–306.

Bartholomäus Wloka and Werner Winiwarter. 2021a. AAA4LLL – Acquisition, Annotation, Augmentation for Lively Language Learning. In *3rd Conference on Language, Data and Knowledge (LDK2021)*.

Bartholomäus Wloka and Werner Winiwarter. 2021b. DARE – a comprehensive methodology for mastering kanji. In *23rd International Conference on Information Integration and Web Intelligence (iiWAS2021)*, pages 427–435.

Yuming Zhai, Aurélien Max, and Anne Vilnat. 2018. Construction of a multilingual corpus annotated with translation relations. In *Proceedings of the First Workshop on Linguistic Resources for Natural Language Processing*, pages 102–111, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

7. Language Resource References

Princeton University. 2012. *WordNet 3.1*. Princeton University, ISLRN 379-473-059-273-1.

YARN is All You Knit
Encoding Multiple Semantic Phenomena with Layers

Siyana Pavlova, Maxime Amblard, Bruno Guillaume
Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France
{firstname.lastname}@loria.fr

Abstract

In this paper, we present the first version of YARN, a new semantic representation formalism. We propose this new formalism to unify the advantages of logic-based formalisms while retaining direct interpretation, making it widely usable. YARN is rooted in the encoding of different semantic phenomena as separate layers. We begin by presenting a formal definition of the mathematical structure that constitutes YARN. We then illustrate with concrete examples how this structure can be used in the context of semantic representation for encoding multiple phenomena (such as modality, negation and quantification) as layers built on top of a central predicate-argument structure. The benefit of YARN is that it allows for the independent annotation and analysis of different phenomena as they are easy to "switch off". Furthermore, we have explored YARN's ability to encode simple interactions between phenomena. We wrap up the work presented by a discussion of some of the interesting observations made during the development of YARN so far and outline our extensive future plans for this formalism.

Keywords: semantics, semantic framework, formalisation, layered semantic representation

1. Introduction

Current semantic representation formalisms can be split into two broad categories - those inspired by Logic (Kamp and Reyle, 1993; Montague, 1970), and those stemming from a graph-based perspective (Banarescu et al., 2013; Abend and Rappoport, 2013; White et al., 2016; Van Gysel et al., 2021). While powerful in terms of encoding, logic-based representations can be difficult to read without prior training in Logic. Graph-based ones, on the other hand, are easier to read, but often lack when it comes to expressing scope or being compositional.

In this work, we aim to find a way to "mediate" between the two and find a representation which is both powerful in terms of encoding as the first group is, but also easier to read and annotate, as the second group is. Thus, we focus on the differences stemming from the logic-based vs graph-based view. Reviews of further differences between various deep-syntax or semantic representation formalisms can be found in the literature (Žabokrtský et al., 2020; Abend and Rappoport, 2017; Pavlova et al., 2023b; Giordano et al., 2023).

We present here the first version of a new semantic representation formalism, YARN (from "laYered meAning RepresentatioN"), with a predicate-argument structure (PA-structure) based on Abstract Meaning Representation (AMR) (Banarescu et al., 2013), and a layered approach to encode semantic phenomena. We provide proof-of-concept examples which demonstrate how the layered structure can be used to encode phenomena such as negation, modality, temporality and quantification, and how they can interact with each other. Considering the interactions between diverse phenomena presents a challenge that existing formalisms do not explicitly address. This question is undeniably complex, yet significant if we aspire to provide a realistic outlook on the practical application of representations. Our initial tests for the cases of modality and temporality show a promising start for YARN's ability to model these.

The main motivation for our approach is to allow the user of the formalism to focus on phenomena that they are interested in exploring, by allowing them to "switch off" the ones they are not interested in as to not clutter the representation. This gives the opportunity to encode specific properties needed for a general interpretation, but still anchor in a global meaning representation. The main contributions of this article are (1) to position the importance of considering the modelling of several semantic phenomena at the same time (2) as well as their interactions in order to (3) propose a rich representation that remains accessible for annotation and use.

In section 2, we present some of the existing semantic representation formalisms which are currently the most developed and have a similar outlook. To fully present the representation power of layers, in section 3, we provide the formal definition for our formalism, followed by annotation examples in section 4. In section 5, we provide a number of discussion points regarding our proposal, as well as aspects concerning the broader topic of semantic representation. This is followed by an outline for our future work in section 6.

5th Workshop on Designing Meaning Representations (DMR 2024) @LREC-COLING-2024, pages 66–76

2. Semantic Formalisms

In this section we outline some existing semantic representation formalisms that we later compare to our proposal. We focus here on AMR as our proposal uses its PA-structure as a base. We then describe Uniform Meaning Representation (UMR) (Van Gysel et al., 2021) as it is an extension of AMR that addresses many of its shortcomings. Finally, we mention Discourse Representation Theory (DRT) (Kamp and Reyle, 1993) as an example of a logic-based formalism.

Abstract Meaning Representation (AMR) (Banarescu et al., 2013) is a formalism that is meant to be simple enough to allow for large-scale annotation. As such, it focuses on the PA-structure of a sentence, annotating core arguments of each predicate according to PropBank's (Palmer et al., 2005) predicates and argument roles, as well as a closed set of non-core roles, to annotate additional arguments such as `time`[1], `location` or `manner`. However, to keep the simplicity, many semantic phenomena such as tense, plurality or scope are not accounted for. AMR has been developed with English in mind and does not claim to be universal. That being said, AMR-annotated datasets exist in multiple languages.

Uniform Meaning Representation (UMR) (Van Gysel et al., 2021) is currently the broadest extension of AMR and can be considered a formalism in its own right. It combines a number of AMR extensions proposed over the years (Donatelli et al., 2018; Pustejovsky et al., 2019) to annotate phenomena such as temporal information, aspect, quantifier scope and co-reference. One of UMR's goals is to keep the simplicity and ease of annotation of AMR, while enriching the set of phenomena it accounts for. UMR is a relatively new formalism and no large corpora exist yet, but annotation work is underway, including annotation procedures for low resource languages.

Discourse Representation Theory (DRT) (Kamp and Reyle, 1993) was introduced with the idea of preserving the principles of compositionality introduced by Montague (Montague, 1970, 1973) while making the representation more accessible. One of the main contributions of DRT is to consider the semantic contribution of an utterance or one of its components as a function that updates the general representation. In this way, it takes into account the process of representation construction. Based on logical representation, it makes the concept of scope explicit by means of boxes containing information in the form of predicates. The logical relationships between them are encoded in such a way that a semantic structure emerges, a structure that is useful, for example, for expressing the accessibility of the variables used. This structure is also extended for discourse representation with SDRT (Asher and Lascarides, 2003). The combination of semantic representation, logical properties and readability makes it a useful formalism. A large corpus of DRT-annotated data exists in the form of the Parallel Meaning Bank (PMB) (Abzianidze et al., 2017). There is also a recent proposal to simplify the notation to foster easier annotation (Bos, 2021).

3. Description

We propose a structure with a central graph, representing the PA-structure, on top of which various layers can be defined. We provide examples to demonstrate how layers can be used to encode semantic phenomena, be it by interacting with nodes in the graph, or between themselves.

We follow the neo-Davidsonian tradition of placing a variable at the centre of the representation, representing the event being described (Davidson, 1967; Parsons, 1990). Our goal is to represent the semantics of an event, encompassing its core PA-structure, and modifiers in a readable and as simple as possible framework.

3.1. Formal Definition

Here is the formal mathematical definition. A YARN is an 8-tuple $< S, V, F, E, \hat{E}, E_{F\hat{E}}, E_{\hat{E}V}, E_s >$ where:

- S and V are sets of vertices
- F is a set of features
- E is a set of edges between pairs of vertices $v_1, v_2 \in V$
- E_{FV}, which we will also call $\hat{E}$, is a set of edges between a feature $f \in F$ and a vertex $v \in V$
- $E_{F\hat{E}}$ is a set of edges between a feature $f \in F$ and an edge $e \in \hat{E}$
- $E_{\hat{E}V}$ is a set of edges between an edge $e \in \hat{E}$ and a vertex $v \in V$
- E_s is a set of edges between a pair of vertices $s_1, s_2 \in S$

We can imagine a layer-based solution using hypergraphs instead, as they are sufficiently expressive, but in order to maintain direct readability we prefer this solution.

[1] When temporal adverbials are present as separate surface tokens.

One way of approaching these definitions is to consider that the central element of the representation is a simple graph around the predicate defining the main event. This gives us a very readable base representation. To avoid making the representation more cumbersome, we don't modify it directly, but allow other information to be added in the form of layers. These new elements lead to the use of new objects that operate either on the graph nodes or on the layer edges.

3.2. From Definition to Semantic View

With the formal definition given, let us look at how YARN can be applied to semantic representation. The vertices from S can be thought of as *event* nodes, with one defined for each event in the text[2].

The vertices of V and edges of E can be thought of as the ones used in the graphs of AMR and AMR-derived representations. Vertices in V represent predicates and concepts. Edges in E represent core argument roles. In this part of the representation, our focus is on the core concept that constitutes the central event. The subcategorisation in a meaning bank helps to identify the mandatory arguments, as for AMRs. Consequently, the representation is lucid and easy to comprehend.

However, concepts representing non-core arguments and their modifiers are not always linked to the main predicate (see Figure 5). This results in elements of V and E making up the PA-structure of the sentence, which is a connected component within the graph, but also a number of (smaller) connected sub-graphs for some of the non-core arguments. Thus, the resulting graph formed by V and E is not necessarily connected.

The vertices of F represent various semantic phenomena, such as temporality, quantification and modality. The vertices are connected by lines that run between the feature nodes and V nodes, resembling strands of yarn. Each phenomenon is assigned a colour to simplify the reading.

The edges in $\hat{E}$, $E_{F\hat{E}}$ and $E_{\hat{E}V}$ are used to represent the linking between the semantic phenomena being annotated and the predicates and concepts of the sentence, as well as between the semantic phenomena themselves. We will see in section 4 how these three different types of edges are used for the different phenomena.

Finally, edges in E_S represent relations between different events, which can be thought of as representing discourse relations.

3.3. Hello World Example

To bridge the formal definition and the pictorial examples that will follow, we will present the first such example also in the mathematical notation that follows naturally from the formal definition.

Let us consider the sentence *"I found a newspaper"*. Its formal representation, ensuing from the YARN definition is the following:

$$\begin{aligned}
&S = \{S_1\} \quad V = \{find{-}01, i, newspaper\}\\
&F = \{temp, quant\}\\
&E = \{(find{-}01, ARG0, i),\\
&\quad (find{-}01, ARG1, newspaper)\}\\
&E_{FV} = \{(quant, \exists, newspaper),\\
&\quad (temp, past, find{-}01)\}\\
&E_{F\hat{E}} = \emptyset \qquad E_{\hat{E}V} = \emptyset \qquad E_s = \emptyset
\end{aligned}$$

We note that for these examples we have chosen PropBank (Palmer et al., 2005) as our predicate sense and argument role bank, utilising PropBank's Frame Files (Choi et al., 2010) to collect the relevant senses and argument roles.

Figure 1 is the YARN graphical representation of the same sentence. The PA-structure of the sentence is a graph which appears in the dotted box, where nodes are predicates and concepts, and edges are relations between a predicate and its arguments. For this sentence, as the vertices of type V, we have the predicate find-01 and two concepts, i and newspaper, representing the two arguments of the predicate. The two arguments are linked to the predicate via two labeled edges of type E, annotating their argument roles as ARG0 and ARG1, respectively. Up to this point (and in this example, but not in general), the PA-structure of YARN coincides with the entire AMR.

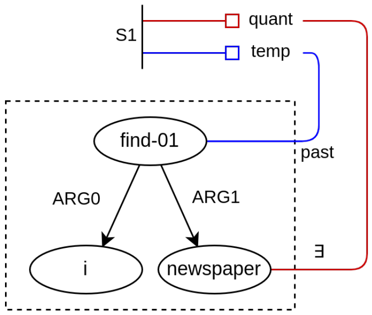


Figure 1: YARN for *"I found a newspaper"*, featuring temporality and quantification.

In addition, we have a vertex $S_1 \in S$, that represents the *event*, to which two features of type F are linked: temp for temporality and quant for quantification. An edge of type $\hat{E}$, labeled past links the temp to the main predicate find-01, indicating the event happened in the past. Another edge of type $\hat{E}$, labeled $\exists$ links quant to newspaper, in-

[2] In this paper, we mainly use simple texts, each containing only one event. See section 5 for a discussion on annotating more complex examples.

dicating existential quantification. For readability's sake, we put a box around the PA-structure, but this is not a part of the formal representation.

Here, we limit the representation to two features to demonstrate their operation. YARN's modularity is advantageous since only specific semantic aspects of modelling can be considered. If the analysis also encompasses others, for example modalities, a modality feature can be added with a new "thread of yarn". We can selectively activate the features that interest us.

4. Towards Multi-Layered Examples

In this section, we demonstrate how the structure described in section 3 can be used to encode various semantic phenomena, with the help of a number of examples. In the following we will concentrate only on a handful of semantic phenomena, but enough to cover the ways to combine different types of elements of YARN. For the sake of clarity, we will use only the graphical representation of the formalism from here on, but the mapping from these representations to the formal one is direct.

Figure 2 is the YARN for *"I couldn't find the newspaper"*. The PA-structure for this sentence is the same as in Figure 1, but differs from the AMR of the sentence, where there are additional nodes and edges to account for the possibility and the negation. Aside from the PA-structure, we have annotated three phenomena: modality, introduced by could, negation introduced by n't, and the temporality of the main predicate. For each, a feature that connects to S_1 is added. Could indicates possibility, so we add an E_F edge linking modal with the corresponding label to the main predicate. The possibility is then negated, with an unlabeled $E_{F\hat{E}}$ edge from neg to the possibility edge. Finally, since the impossibility was in the past, we add an $E_{F\hat{E}}$ edge from temp to the possibility edge, with a label past.

Temporality classes include past, present and future for now and modality classes: possibility and necessity. We kept this simple as the choice of classes for each phenomenon is not the focus of this work. These will be extended and made to account for different granularities across languages via lattices (Van Gysel et al., 2019). This approach has already been adopted for meaning representations by UMR.

To discuss $E_{\hat{E}V}$, we will use the example in Figure 3. This very simple example helps us to showcase how YARN deals with a classical logical issue, quantification. Figure 3 is one of the possible representations for the sentence *"Every cow ate an apple"*. This sentence has two quantifiers - universal for cow and existential for apple, thus giving rise to scope ambiguity. Two readings exist: one where *"every cow"* takes wider scope, encoding the meaning where every cow ate a different apple, and one where *"an apple"* takes wider scope - where all cows ate the same apple. The representation in Figure 3 is for the latter. Here, aside from the temp feature, we have introduced a quant feature, linked as usual to S_1. An edge of type E_F links quant to the wider-scope taking entity, namely apple, labeled with the appropriate quantifier, in this case $\exists$. Finally, an edge of type $E_{\hat{E}V}$ is introduced linking the $\exists$ edge to the narrower scope entity cow. The appropriate label, $\forall$, is given to this edge. Thus, when annotating multiple quantifiers in a representation, we introduce a quant feature, then link it to the outermost scope-taking argument. Moving inwards, each argument is linked to the previous scope-defining edge.

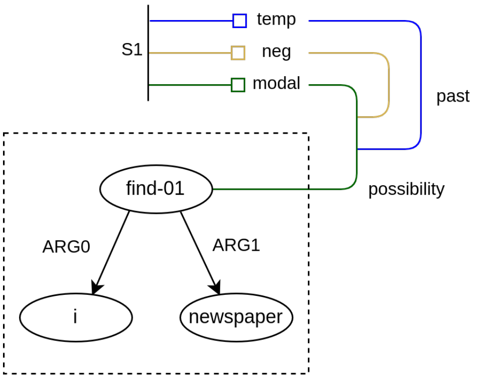


Figure 2: YARN for *"I couldn't find the newspaper"*, featuring temporality, negation and modality.

A key issue in representing quantifiers semantically is the potential for combining scopes in various orders. The fundamental inquiry is whether every cow consumes an apple that is its own, or whether every cow consumes the same apple. Undoubtedly, pragmatics directs us towards a preferred interpretation for cows and many apples. Figure 4 provides an illustration of how quantifier scopes can be reversed. It appears that the type of link used for the quantifiers has changed, with the link for $\exists$ deriving from an element of type $E_{\hat{E}V}$ and the link for $\forall$ from type E_F. The entity now taking wider scope is the cow. The continuous link in the graphical representation represents the wider scope, while the link starting from the junction circle represents narrow scope. This approach fully utilises the expressiveness of YARN enabling the retention of readability whilst explicitly addressing logical constraints.

To represent some non-compulsory arguments, such as manner or location, we propose a solution as the one in Figure 5. Here, we have the representation of the sentence *"Every cow ate an apple in the garden"*. In addition to the two predicate-specific arguments of eat-01, an op-

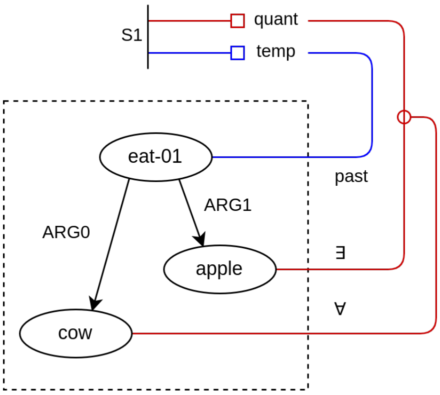


Figure 3: YARN for *“Every cow ate an apple”*, featuring quantification and temporality. Reading where all cows ate the same apple.

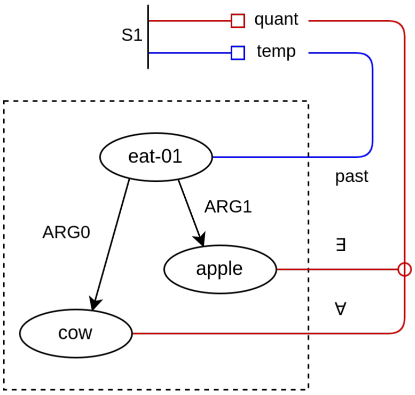


Figure 4: YARN for *“Every cow ate an apple”*, featuring quantification and temporality. Reading where every cow ate a different apple.

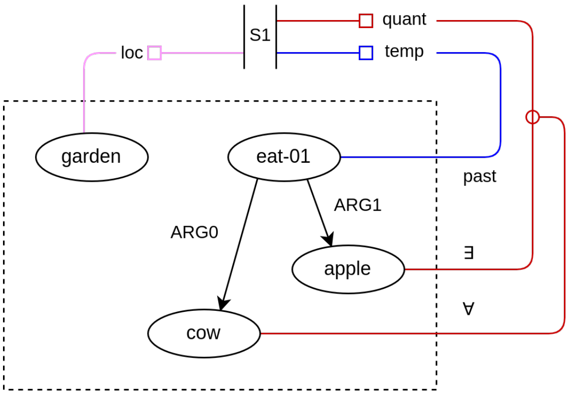


Figure 5: YARN for *“Every cow ate an apple in the garden”*, featuring an additional argument for location.

tional argument for location, namely *“in the garden”*, is specified. In AMR, optional arguments are attached to the predicate using the so called non-core roles in the same manner as predicate-specific ones. In our proposal, we annotate some of them as separate nodes (or subgraphs, in the case of more complex modifiers) that appear in the same box as the main predicate. To specify that argument's role, we introduce a feature of type F, and an unlabeled edge of type E_F from the feature to the argument. In the example in Figure 5, a new feature loc is added that links S_1 to the node garden that has been added to the predicate box. The same can be done for other kinds of modifiers that are typically annotated with non-core roles in AMR.

In the preceding examples, we demonstrated how to formulate the control component that characterises the event, how to append non-compulsory parameters, and how the yarn principle straightforwardly encompasses distinct semantic phenomena, by integrating aspects of scope. The addition of a variable $s \in S$, which stands for the event, is a beneficial realisation for modelling other occurrences, including those that are conventionally encountered in discourse representation. Figure 6 is a sample representation of the sentence *“I entered the room, because the phone rang”*, where we have a causal relation between the two events *“enter”* and *“ring”*. For simplicity, we have chosen not to show the quantification annotation here. However, this is entirely possible and would result in a more extensive sample, allowing the reader to select certain features for analysis. We introduce an edge of type E_S, labeled CAUSE, between the two *event* nodes S_1 and S_2. This example shows that this representation also provides a solution for annotating discourse relations, taking the representation beyond semantics. Discussion of labelling the links between the S type elements that construct a higher level structure is beyond the scope of this work. Common discourse theories, such as SDRT (Asher and Lascarides, 2003) or RST (Mann and Thompson, 1986), can be utilised. YARN remains theory agnostic. In YARN, variables can represent elementary discourse units (EDUs) which creates a structure that covers the entire document, similar to SDRT. Alternatively, we can introduce relations between specific elements, as is done in RST.

Having seen the definition of the structure in subsection 3.1, and the examples above demonstrating how each can be used in the context of semantic representation, we sum up the characteristics for each element of our YARN 8-tuple in the context of semantic annotation. These elements demonstrate the technical nuances of formalisation, which can be linguistically interpreted.

- Edges in E and E_S are directed. For the rest of the edges, while there is an implicit direction - from a feature $f \in F$ towards either a vertex or another edge, or from an edge in E_{FV}, $E_{F\hat{E}}$ or $E_{\hat{E}V}$ towards a vertex, there is

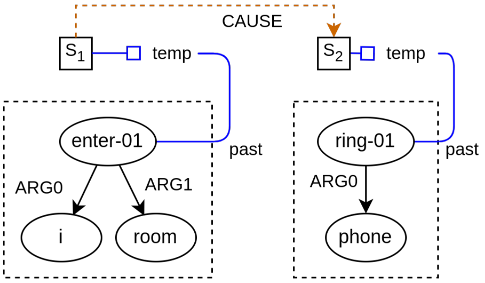


Figure 6: YARN for *"I entered the room, because the phone rang"*, featuring a discourse relation between two events.

no need to draw it.

- Not all relations need to be labeled, only the information needed to disambiguate the interpretation is required.
- Each element f of F is linked to the *event* node of the representation, which means that it can be considered as the reification variable on which all the properties are applied.
- V is not a closed set. While users can choose the specific lexicon for the predicate senses and even the concepts, there is no restriction to do so - predicate senses and concepts can be used freely as the context in which the formalism is used requires it.
- F is a closed set of semantic phenomena. We have only briefly addressed some of these principles, deferring in-depth discussion to the future.
- Labels in E are a closed set – they can come from semantic role lexicons like VerbNet (Kipper et al., 2008), or be the set of core + noncore roles from AMR, just as two possible examples.
- Labels in E_F are a closed set, made up of the subsets defined for each feature. For these, we'll use lattices for each feature, inspired by (Van Gysel et al., 2021).
- Quantifiers are expressed as labels on $\hat{E}$ or $E_{\hat{E}V}$ edges. In the examples, we have solely employed common quantifiers, but it is possible to expand the list to encompass additional forms of more precise quantification.

These formal characteristics are crucial since they enable us to have a controlled representation that can be projected onto a logical representation by means of a simple algorithm, as in the case of DRT, while at the same time offering an adaptation to the needs of the linguistic representation without mixing up the different elements, allowing us to focus on particular points.

5. Discussion

Here we discuss some of the observations we made while designing YARN. These concern features of the formalism itself and more general observations and questions about the kinds of phenomena we may want to include moving forward.

5.1. Yarn interactions

The introduction of these "strands of yarn" gives us a flexible structure for the representation and introduces a level where new interactions are possible, either through their non-explicitation with the subspecification or through the swapping of relations.

Some semantic representation formalisms, such as Minimal Recursion Semantics (Copestake et al., 2005) allow for underspecification, for example in the case of scope ambiguity. While not present in the current version, as part of future work, we intend to offer the possibility for underspecification in the representation. In Figure 7 we provide a graphical representation of one possible solution for scope ambiguity/underspecification. As can be seen in the figure, the scope precedence between *"an apple"* and *"every cow"* has not been resolved. When the precedence becomes apparent from context, the representation can be updated in order to accommodate for that. As part of our future work, we intend to formalise this new structure, and investigate if the same one can be used for other types of underspecification.

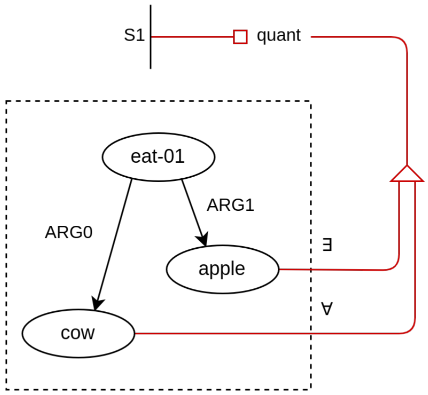


Figure 7: *"Every cow ate an apple"* - YARN with underspecified scope.

In the example in Figure 2 we saw that the layers belonging to different phenomena can be stacked on top of one another. This property is not commutative. In Figure 2, modality is applied to the main predicate, followed by a negation applied to the modality in order to encode *"couldn't [do* *something]"*. Compare this to the example in Figure 8, where the modality and negation have been swapped. If we try to build the meaning from the representation, we end up with *"Maybe/[It is possible that] I did not find the newspaper"*.

As our focus lies on the formal aspects of the framework, we will reserve the analysis and discussion of the stands of yarns' interactions for future research. For instance, we have yet to exhaustively investigate how modality and temporality interact and significantly alter the resulting interpretation.

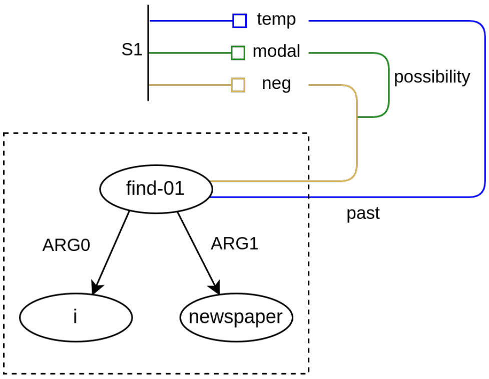


Figure 8: YARN of *"Maybe I did not find the newspaper"*.

Our goal with this proposal is to really separate the various semantic phenomena into different layers, and the focus is to make them easier to see at first glance. This is the reason why we prefer to disconnect optional arguments from the core PA-structure where that is linked to semantic phenomena. However, the mapping between this disconnected version and the AMR-style version is preserved, at least for what we have tested so far: simple combinations of the modifiers for time and location. Transformation functions between our "disconnected" version and the "connected" AMR-style one will be the subject of future work, allowing an easier transition between our formalism and AMR-based structures.

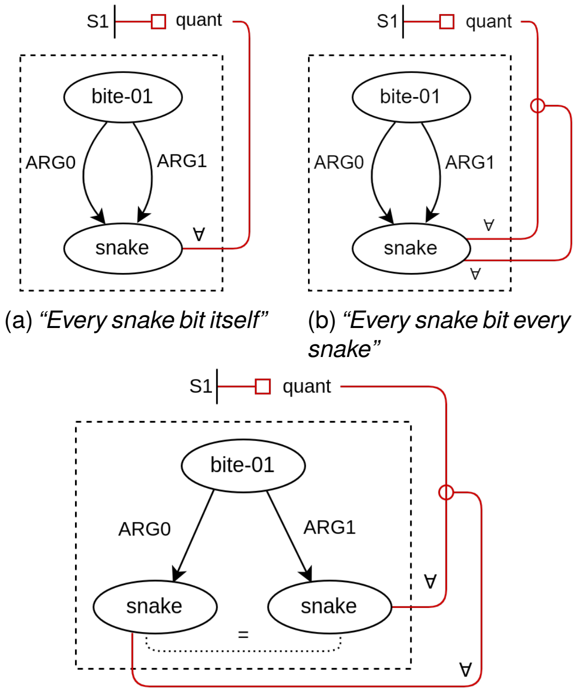
5.2. Comparison to other formalisms

AMR is the basis for our representation, but, as we have shown, YARN differs from it substantially in the way it encodes semantic phenomena. Thanks to this, we are able to achieve a richer representation, without making the reading too complex.

Bos (2020) points out that the proposal to extend AMR for quantification and scope presented in (Pustejovsky et al., 2019) suffers from the so called *bound variable problem*: the way sentences such as *"Every snake bit itself"* are represented, can be interpreted also as *"Every snake bit every snake"*. Since UMR (Van Gysel et al., 2021) implements the above-mentioned extension, it faces the same issue. To tackle this, we need different representations for the two sentences. The YARN for *"Every snake bit itself"* is straightforward: in Figure 9a, the presence of a single quantifier indicates that the same entity participates both as an ARG0 and ARG1 in each occurrence of the biting event. Thus, for *"Every snake bit every snake"*, we perforce need two quantifiers. Figure 9b shows such a representation. However, we face another issue here: we cannot tell whether the wide-scope taking entity is the *biter* or the *bitee*. In the example with two universals, however, explicit resolution of the ambiguity is not required: the two options have different representations, but, when resolved, the interpretation is the same. However, in a case where we have this same representation but different quantifiers, this information is necessary. One possible solution for the same case but with different quantifiers, would be to include the name of the argument to which it is applied to the quantification edge, for instance $\forall$: ARGO. It does, however, still seem strange, to have two quantifiers pointing to the same entity (or set of entities). What is more, if we explore the other three quantifier combinations for two entities, we see that two universals is a special case: in all other cases, it is more natural to have separate entities for each participant. Thus, we propose the solution in Figure 9c. Here, we split the *biters* and *bitees* into two separate nodes and acknowledge, via a new type of edge (the dotted link in the figure) that in this special case it happens that the two nodes refer to the same set of entities, via the $=$ sign. This new type of relation is yet to be formalised in the next version of YARN. This can be useful not only here, but also for linking co-referents (see Figure 10). Lastly, the representation in Figure 9b can still remain, purely for ease of readability, with the caveat that it is simply a visualisation, equivalent to the one in Figure 9c and in the background, the latter is the canonical form.

Aside from this, for formalisms where all events remain in the same graph, the representation becomes difficult to follow, especially for longer texts, as can be seen with UMR (Zhao et al., 2021). We believe our solution to represent each event as its own substructure makes our representation more easily readable, even for larger texts, making it easy to spot the phenomena applying to each event and the interactions between events thanks to the discourse-style relations between them.

This is illustrated by the annotation in Figure 10 for the sentence *"I couldn't find the newspaper until you told me where it suddenly appeared"*. Here, we have three events: *finding*, *telling* and *appearing*, each represented by its own S-type node. Thanks to these, we can easily differentiate the features that apply to each event, and also track



(a) *"Every snake bit itself"*

(b) *"Every snake bit every snake"*

(c) *"Every snake bit every snake"*

Figure 9: YARN for snake examples

the relationships between events. This example also illustrates that while subordinate clauses introduced by subordinating conjunctions such as *"before"*, *"until"* or *"because"* can be modelled by edges from E_S (as we see between S1 and S2), where a predicate permits it, they can be modelled by argument roles from E (as with tell-01 from S2 and the empty node marking the *location* from S3). Finally, by using the same type of edges as the ones proposed for the equality between sets in Figure 9c, we can also model co-reference, as shown by the links between the i's in S1 and S2, and newspaper in S1 and it in S3.

It is worth noting that YARN also draws inspiration from DRT, although it is not a direct representation of it. As a result, the algorithmic principle is used to convert the representation into a standard logical formula, which induces the same structure as DRT. The crucial elements primarily lie in "stands" of quantification. The conversion from one to the other is a task for future work. Furthermore, thanks to the S nodes, we can imagine an expansion of the depiction from a standpoint comparable to SDRT's expanding of DRT.

5.3. Some broader questions

We can have a broader discussion on what constitutes an event and how events are deduced from the surface form of a sentence. If we take the sentence from the WSJ (Paul and Baker, 1992), *"Edmond Pope tasted freedom today for the first time in more than eight months."*, we have the main event *E*, *"tasted [freedom]"*, but also a reference point to something that happened *"more than eight months ago"*. Thus, we may ask whether apart from *E*, we also have another (possibly static) event, *E'* of *"having tasted freedom"*, or *"having been free"* more than eight months ago. We may argue whether such implication (the one of *E'*) is or should be part of the semantic representation of the sentence, or whether we should only annotate events that appear explicitly in the sentence.

Our annotation experiments so far demonstrate that it is easy to extend the formalism with new semantic phenomena. Adding a new one so far has consisted in adding a new feature to F and deciding on the appropriate type of relations to use for edges of that layer. To test our proposal, we are currently analysing sentences in English from the Parallel Universal Dependencies (PUD) corpus[3]. Although our proposal is currently robust, we may encounter issues when annotating more complex phenomena or sentences. It is yet to be determined, following the expansion of the range of observable occurrences, whether working on a larger dataset will help to test the ease of use of the framework. The annotation of a substantial corpus will enable us to assess the capabilities of YARN.

6. Conclusion and Future Work

In this paper, we presented the first version of YARN, a proposal for encoding multiple semantic phenomena with layers. The framework differs from others in that it maintains a logical structure, while remaining clear to the reader. The incorporation of diverse levels allows for the comprehensive modelling of various phenomena, whilst still maintaining their distinctiveness and potential interconnections.

We have shown, through examples, that our initial annotations show a promising structure that manages to encode difficult phenomena and keep the representation visually simple. Analysis is further aided by the fact that "switching off" layers is straightforward. We have highlighted interesting discussion points that were raised during the design of our formalism, and have outlined the future work directions for this project.

As we have shown, the thus proposed structure, YARN, is capable of representing a range of semantic phenomena, namely: temporality, modality, negation, quantifier scope. While not presented here due to space limits, we have also tested definiteness, number and questions. In the preceding section, we presented various view-

[3] https://github.com/UniversalDependencies/UD_English-PUD

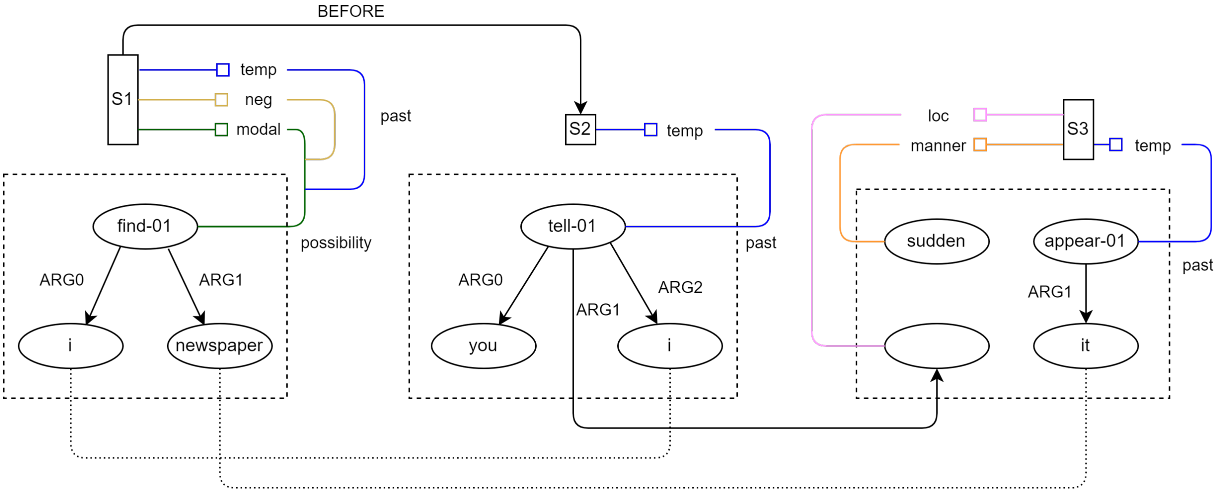


Figure 10: YARN of *"I couldn't find the newspaper until you told me where it suddenly appeared"*.

points on the evolution of specific phenomena. We now go back to more general aspects.

As a first step, in our future work, we plan to add more phenomena to the formalism, such as comparison, gender, predicates whose arguments are events (such as "begin", "stop"), etc. For each phenomenon, a set of possible classes will be defined. We do not intend to limit the classes, but rather allow lattices as presented in (Van Gysel et al., 2019) and used in UMR in order to enhance comparison between languages without limiting the possible classes to those available in a specific language.

In parallel to this, we will formalize annotation guidelines and develop annotation tools, with the help of which to carry out annotation experiments.

As mentioned earlier, one of our goals is to provide a formalism where "switching off" layers is simple, which is a major difference from others such as UMR. This is straightforward in cases where layers do not interact with each other (as in Figure 1). However, in more complex cases such as in Figure 2, the process is not straightforward. Removing the `temp` layer would not necessarily affect the `modal` layer as it is attached on top of it. However, what would it mean for the `temp` feature and the interpretation of the whole representation if only the `modal` feature were to be removed? Understanding this interaction and defining procedures on how to "switch off" a layer that interacts with other layers will be the subject of another future work.

As we want our representation to be able to "communicate" with both logic-based and established graph-based formalisms, we envision two further future work directions: (1) make explicit the formal procedure to convert a YARN into first-order logic and vice-versa, and (2) creating transformation systems between ours and other graph-based formalisms, in the spirit other transformation-based comparison works (Hershcovich et al., 2020; Pavlova et al., 2022, 2023a).

Finally, we want to propose a textual representation format for YARN, in the spirit of the PENMAN notation (Matthiessen and Bateman, 1991), widely used for AMR, and AMR-derived formalisms. We expect having such a representation will be useful for developing parsing algorithms for our formalism, both with symbolic and hybrid approaches.

7. Ethical Considerations

While Universality is one of the desired features for the presented meaning representation, we note that there is likely an inherent bias towards phenomena which are more prevalent in occidental linguistic culture, and English in particular, which is the main language we have used so far for YARN's development. While we have not had the chance to do this for the current version of the formalism, we acknowledge that a more thorough study and discussion of non-occidental languages is necessary for a less biased representation. This is further affected by our use of PropBank, a sense lexicon, an equivalent of which is not available for the majority of world languages. Thus, we also need to employ strategies for either a resource agnostic resource development or follow UMR's steps in proposing strategies on how to build and extend such resources for low-resource languages.

Acknowledgments

We would like to thank the anonymous reviewers for their feedback and comments. Part of this work has been funded by *Agence Nationale de la Recherche* (ANR, fr: National Research Agency), grant number ANR-20-THIA-0010-01.

8. Bibliographical References

Omri Abend and Ari Rappoport. 2013. Universal Conceptual Cognitive Annotation (UCCA). In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 228–238, Sofia, Bulgaria. Association for Computational Linguistics.

Omri Abend and Ari Rappoport. 2017. The state of the art in semantic representation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 77–89, Vancouver, Canada. Association for Computational Linguistics.

Lasha Abzianidze, Johannes Bjerva, Kilian Evang, Hessel Haagsma, Rik van Noord, Pierre Ludmann, Duc-Duy Nguyen, and Johan Bos. 2017. The Parallel Meaning Bank: Towards a multilingual corpus of translations annotated with compositional meaning representations. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 242–247, Valencia, Spain. Association for Computational Linguistics.

Nicholas Asher and Alex Lascarides. 2003. *Logics of Conversation*. Studies in Natural Language Processing. Cambridge University Press.

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 *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria. Association for Computational Linguistics.

Johan Bos. 2020. Separating argument structure from logical structure in AMR. In *Proceedings of the Second International Workshop on Designing Meaning Representations*, pages 13–20, Barcelona Spain (online). Association for Computational Linguistics.

Johan Bos. 2021. Variable-free discourse representation structures. *Semantics Archive*.

Ann Copestake, Dan Flickinger, Carl Pollard, and Ivan A Sag. 2005. Minimal recursion semantics: An introduction. *Research on language and computation*, 3(2):281–332.

Donald Davidson. 1967. The logical form of action sentences. In Nicholas Rescher, editor, *The Logic of Decision and Action*, pages 81–95. University of Pittsburgh Press.

Lucia Donatelli, Michael Regan, William Croft, and Nathan Schneider. 2018. Annotation of tense and aspect semantics for sentential AMR. In *Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018)*, pages 96–108, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Bastien Giordano, Cédric Lopez, and Immeuble Le. 2023. Mr4ap: Meaning representation for application purposes. In *The 15th International Conference on Computational Semantics (IWCS 2023)*.

Daniel Hershcovich, Nathan Schneider, Dotan Dvir, Jakob Prange, Miryam de Lhoneux, and Omri Abend. 2020. Comparison by conversion: Reverse-engineering UCCA from syntax and lexical semantics. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2947–2966, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Hans Kamp and Uwe Reyle. 1993. *From discourse to logic: Introduction to modeltheoretic semantics of natural language, formal logic and discourse representation theory*. Dordrecht. Kluwer.

Karin Kipper, Anna Korhonen, Neville Ryant, and Martha Palmer. 2008. A large-scale classification of english verbs. *Language Resources and Evaluation*, 42:21–40.

William C. Mann and Sandra A. Thompson. 1986. Relational propositions in discourse. *Discourse processes*, 9(1):57–90.

Christian MIM Matthiessen and John A Bateman. 1991. Text generation and systemic-functional linguistics: experiences from english and japanese. *Communication in Artificial Intelligence Series*, 19(1).

Richard Montague. 1970. English as a formal language. *Logic and philosophy for linguists*.

Richard Montague. 1973. The proper treatment of quantification in ordinary english. In *Approaches to natural language: Proceedings of the 1970 Stanford workshop on grammar and semantics*, pages 221–242. Springer.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Terence Parsons. 1990. *Events in the Semantics of English: A Study in Subatomic Semantics*. MIT Press.

Douglas B. Paul and Janet M. Baker. 1992. The design for the Wall Street Journal-based CSR corpus. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*.

Siyana Pavlova, Maxime Amblard, and Bruno Guillaume. 2022. How much of UCCA can be predicted from AMR? In *Proceedings of the 18th Joint ACL - ISO Workshop on Interoperable Semantic Annotation within LREC2022*, pages 110–117, Marseille, France. European Language Resources Association.

Siyana Pavlova, Maxime Amblard, and Bruno Guillaume. 2023a. Bridging Semantic Frameworks: mapping DRS onto AMR. In *The 15th International Conference on Computational Semantics (IWCS 2023)*, Nancy, France.

Siyana Pavlova, Maxime Amblard, and Bruno Guillaume. 2023b. Structural and global features for comparing semantic representation formalisms. In *The 4th International Workshop on Designing Meaning Representation*.

James Pustejovsky, Ken Lai, and Nianwen Xue. 2019. Modeling quantification and scope in Abstract Meaning Representations. In *Proceedings of the First International Workshop on Designing Meaning Representations*, pages 28–33, Florence, Italy. Association for Computational Linguistics.

Jens E. L. Van Gysel, Meagan Vigus, Pavlina Kalm, Sook-kyung Lee, Michael Regan, and William Croft. 2019. Cross-linguistic semantic annotation: Reconciling the language-specific and the universal. In *Proceedings of the First International Workshop on Designing Meaning Representations*, pages 1–14, Florence, Italy. Association for Computational Linguistics.

Jens EL Van Gysel, Meagan Vigus, Jayeol Chun, Kenneth Lai, Sarah Moeller, Jiarui Yao, Tim O'Gorman, Andrew Cowell, William Croft, Chu-Ren Huang, et al. 2021. Designing a uniform meaning representation for natural language processing. *KI-Künstliche Intelligenz*, 35(3-4):343–360.

Aaron Steven White, Drew Reisinger, Keisuke Sakaguchi, Tim Vieira, Sheng Zhang, Rachel Rudinger, Kyle Rawlins, and Benjamin Van Durme. 2016. Universal decompositional semantics on universal dependencies. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1713–1723.

Zdeněk Žabokrtský, Daniel Zeman, and Magda Ševčíková. 2020. Sentence meaning representations across languages: What can we learn from existing frameworks? *Computational Linguistics*, 46(3):605–665.

Jin Zhao, Nianwen Xue, Jens Van Gysel, and Jinho D. Choi. 2021. UMR-writer: A web application for annotating uniform meaning representations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 160–167, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

9. Language Resource References

Choi, Jinho D. and Bonial, Claire and Palmer, Martha. 2010. *PropBank Frame Files*. distributed by Linguistic Data Consortium. PID https://github.com/propbank/propbank-frames/.

Argument Sharing in Meaning Representation Parsing

Maja Buljan, Lilja Øvrelid, Stephan Oepen
Language Technology Group, University of Oslo
{majabu, liljao, oe}@ifi.uio.no

Abstract

We present a contrastive study of argument sharing across three graph-based meaning representation frameworks, where semantically shared arguments manifest as reentrant graph nodes. For a state-of-the-art graph parser, we observe how parser performance – in terms of output quality – covaries with overall graph complexity, on the one hand, and presence of different types of reentrancies, on the other hand. We identify common linguistic phenomena that give rise to shared arguments, and therefore node reentrancies, through a small-case and partially automated annotation study and parallel error anaylsis of actual parser outputs. Our results provide new insights into the distribution of different types of reentrancies in meaning representation graphs for three distinct frameworks, as well as on the effects that these structures have on parser performance, thus suggesting both novel cross-framework generalisations as well as avenues for focussed parser development.

1. Introduction

Over the past decade, there has been increasing interest in parsing into graph-based meaning representations, with a growing field of research across different linguistic traditions and frameworks for meaning representation in terms of labelled graphs. A range of parsing systems and approaches have been developed, as well as various frames of in-depth analyses into particular features and challenges of the task and individual frameworks. Unlike widely used representations of syntactic structure in the form of rooted trees, common meaning representation frameworks employ *general graphs*, which makes parsing into these representations more complex, due to, among other features, fewer structural constraints on elements of the graph and on correspondences to the underlying input string (“anchoring”), as well as, of course, the presence of graph nodes with an in-degree greater than one (henceforth “reentrancies”).

We follow in this line of research by expanding the methodologies proposed in Buljan et al. (2022), and based on English data and systems featured in the 2020 Shared Task on Cross-Framework Meaning Representation Parsing (Oepen et al., 2020). We focus on PERIN (Samuel and Straka, 2020), the top-performing parsing system in the shared task, and conduct a contrastive error analysis over three frameworks (elaborated in Section 2) to identify common parsing errors, with a view to devising potential parser improvements.

Our research shows an unexpected outlier to the widely accepted wisdom that parsing accuracy deteriorates with growing structural complexity. In an effort to identify potential explanations of this behaviour, we look into the phenomenon of argument sharing in meaning representation graphs, giving rise to the aforementioned reentrant structures. Following the methodology of Szubert et al. (2020) and extending it to the two other frameworks, we attempt to set a foundation for expanding our understanding of the effects of framework design decisions, which will eventually allow for informing future annotation, as well as more targeted parser development.

Apart from these empirical findings, the technical contributions of this paper are: a substantial augmentation of `mtool`, the open-source graph analysis and scoring tool first introduced in the 2019 MRP (Meaning Representation Parsing) shared task (Oepen et al., 2019), which enables quantitative and qualitative analysis of reentrancies and their various subtypes; and a small-scale manual reentrancy annotation effort over gold standard data used for parser development in the shared task. Both contributions will be released openly upon publication.

The paper is organised as follows: Section 2 gives a broad summary of the methodological and technological context of our work; Section 3 describes our approach to parser performance analysis, presents the results, and motivates further investigation. In Section 4, we look into the underlying framework properties and how they inform our error analysis. Section 5 discusses different linguistic causes of reentrancy structures in meaning representation graphs, describes the setup of our pilot annotation effort, and presents its findings. Finally, Section 6 concludes the paper, and discusses pertinent next steps.

2. Background

The MRP 2019 and 2020 shared tasks on cross-framework meaning representation parsing were organised with the goal of advancing the state-of-the-art in parsing into graph-based representations of sentence meaning (Oepen et al., 2019, 2020). The task focussed on five semantic graph frameworks, and required participants to develop systems that predict sentence-level meaning representations for all five frameworks in parallel.

Of the five frameworks present in the shared task, we narrow our focus (and use development data pertaining) to three frameworks embodying distinct approaches to meaning representation, differing in their level of abstraction from the underlying surface string, as well as in formal construction and linguistic assumptions. The three frameworks are exemplified in Figures 1, 2, and 3 with the sentence *"Pierre Vinken, 61 years old, will join the board as a nonexecutive director Nov. 29."* (Oepen et al., 2020).

Elementary Dependency Structures (EDS; Oepen and Lønning, 2006, Figure 1) encode sentence meaning in an unordered semantic graph that is derived from the underspecified logical forms of the English Resource Grammar (Flickinger et al., 2017; Copestake et al., 2005). EDS nodes are explicitly anchored onto substrings of the underlying sentence, but these do not correspond one-to-one to surface lexical units, while edge labels denote argument positions into semantic predications.

Prague Tectogrammatical Graphs (PTG; Zeman and Hajic, 2020, Figure 2) present a conversion from the multi-layered (and somewhat richer) annotations in the tradition of Prague Functional Generative Description (FGD; Sgall et al., 1986), as adopted (among others) in the Prague Czech–English Dependency Treebank (PCEDT; Hajič et al., 2012). PTG nodes are mostly anchored to surface lexical units, but allowing for empty ("generated") nodes and discontinuous anchoring Edges in PTG denote fine-grained labelled relation types ("functors").

Abstract Meaning Representation (AMR; Banarescu et al., 2013, Figure 3), in contrast, makes no explicit connection between the surface sentence and elements of the graph, and is therefore considered unanchored (or free of specific assumptions about derivation and composition). Graph nodes are content words most frequently normalised to verbal senses, and edges are labelled with argument positions or more specific semantic relations, including e.g. fine-grained annotations of named entities and some lexical decomposition.

The MRP 2020 English validation data for the cross-framework track, from which we draw data for our work, comprises gold annotated meaning representation graphs of sentences, counting 3302 datapoints for EDS, 1664 for PTG, and 3560 for AMR, respectively (Oepen et al., 2020).

When analysing parser performance, we focus on the top-scoring, state-of-the-art parser from the MRP 2020 shared task: PERIN (Samuel and Straka, 2020). The PERIN parser is a general neural network architecture for learning to predict the mapping from surface strings to various types of linguistic structure in the form of general graphs. Using an XLM-R and transformer-based encoder-decoder architecure, the parser is language- and framework-agnostic, and therefore applicable across different meaning representation frameworks and languages with the adjustment of pre- and post-processing steps. It also uses a novel permutation-invariant approach to parallel graph node prediction, which is well suited to the task of predicting orderless semantic graphs. Furthermore, as PERIN is not a seq2seq model, but based on a specialized node, edge, and label prediction architecture, there is room for follow-up engineering in light of findings such as those presented in this study.

In the broader sphere of parsing data analysis, we build on methodologies inspired by contrastive approaches introduced in, among other works, McDonald and Nivre (2011) and Kulmizev et al. (2019) for dependency treebanks and parsers. We follow and expand upon the quantitative and qualitative approach to error analysis in MRP outlined in Buljan et al. (2020, 2022). We also look to Szubert et al. (2020) for a discussion of reentrancies in AMR and underlying linguistic phenomena.

We report performance using `mtool`[1], the cross-framework graph analyser used in the MRP shared tasks. Other notable framework-specific graph similarity metrics are discussed by Cai and Knight (2013) and Opitz (2023).

3. Analysing Parser Performance

Following the methodology outlined by Buljan et al. (2022), we examine the performance of the PERIN parser on the MRP 2020 shared task, retrained on the official training data, and using the validation data for our study. To make our results robust to fluctuation that could arise from random initialization, we set out to compare five separate training and testing runs. By and large, our observations are stable across all runs.

Graph complexity We begin by dividing the data into ten decile bins, according to sentence-level graph complexity in terms of the number of nodes. Following the official metric of the MRP shared tasks (Oepen et al., 2019, 2020), we focus on the micro-average F_1 score over tuple types that encode various graph properties, where Buljan et al. (2022) observe that it can be beneficial

[1] https://github.com/cfmrp/mtool

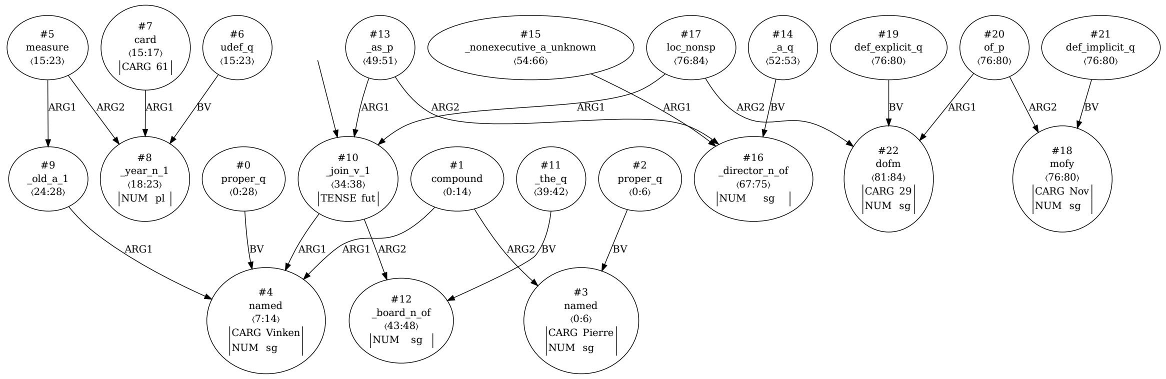


Figure 1: EDS semantic graph for the running example.

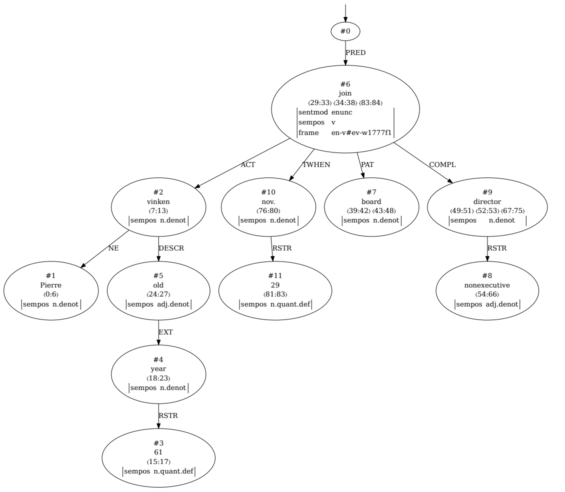


Figure 2: PTG semantic graph for the running example.

to tease apart two distinct subtasks in graph prediction: (a) predicting graph nodes and their decorations, e.g. labels and other node-local properties vs. (b) core graph structure in terms of (labelled) edges and identification of the top node. Figure 4 reports PERIN performance across frameworks and decile bins. The top plot of each framework-specific group charts the overall MRP F_1 score; the middle figure charts F_1 considering only node decoration[2]; and the bottom figure charts F_1 over structural properties only (root nodes (tops) and edges).

We observe a drop in parser performance in the overall F_1 score charts, for PTG and AMR particularly, correlated with rising graph complexity. As discussed in Section 1 above, this is expected behaviour given an assumed correlation between output structure size, sentence length, and related complexity of the parsing problem. However, EDS subverts these expectations, showing instead only a drop of a couple percentage points in the first

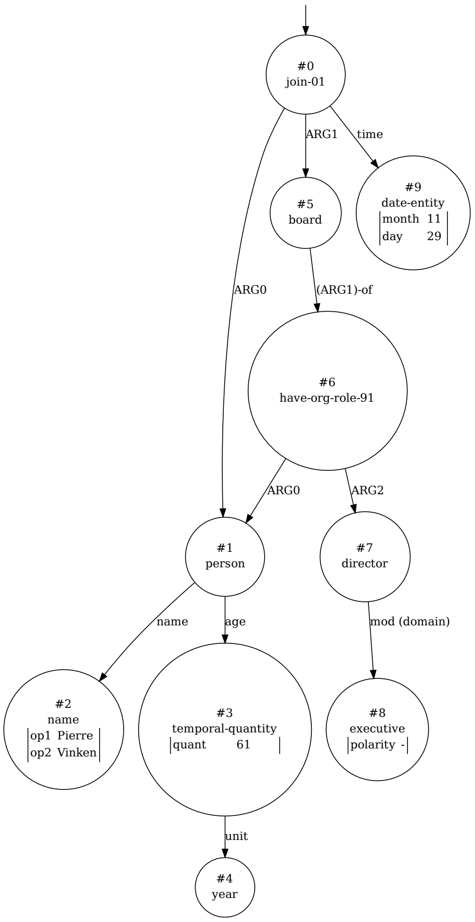


Figure 3: AMR semantic graph for the running example.

[2] Prediction of node anchoring is disregarded, for the sake of result comparability, as it is not applicable to AMR nodes.

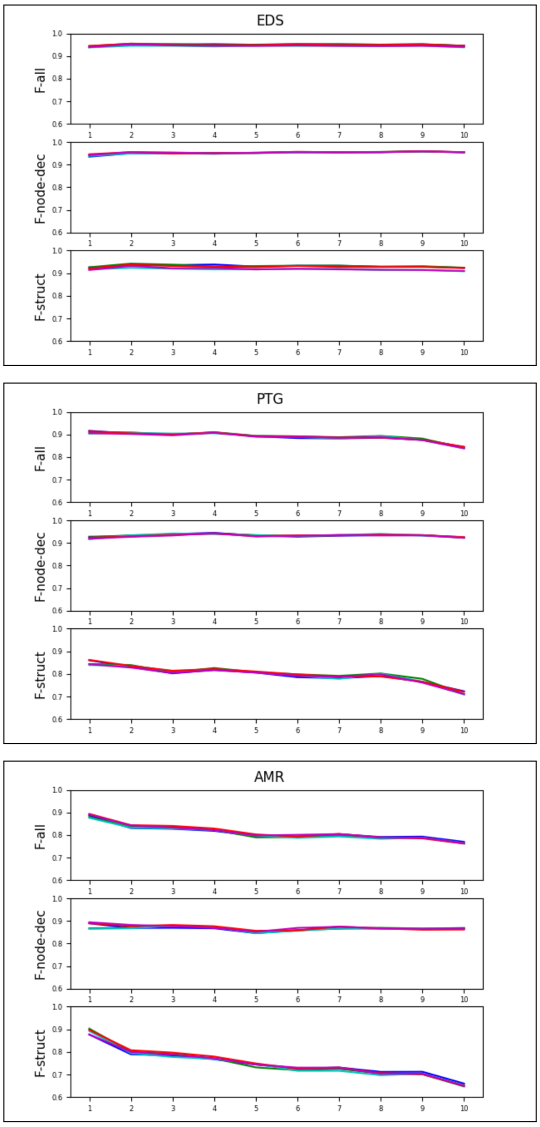


Figure 4: Average parser performance per complexity bin, in terms of overall F_1 (top in each graph), F_1 for node-local properties only (middle), and F_1 for structural properties only (bottom), across the three frameworks.

and last bins – which represent somewhat uneven groups of very short and very long sentences.

Node-local properties The performance over all three frameworks is fairly consistent across the complexity bins when evaluating only on node decoration (node labels and properties). This indicates that node-local information may be easier (for PERIN) to predict with consistent quality.

Structural properties When we examine the parser performance on structural properties of the graphs (edges and root nodes), a clearer picture emerges of the performance drop with greater graph complexity for PTG and AMR. Compared to a drop of 15 and 9 percentage points, respectively, between the highest and lowest performing bin in PTG and AMR overall, the drop in performance is 26 and 16 percentage points for the structural properties. Again, though, EDS remains the outlier, with relatively consistent scores across the bins (within 3-5 percentage points), and moderate divergence between the different training and scoring runs.

	EDS	PTG	AMR^{-1}
Average Nodes / Graph	23.4	17.9	10.4
Edge Labels	10	67	84
$\%_g$ Rooted Trees	0.3	23.9	27.0
$\%_g$ Treewidth One	66.9	23.9	53.7
Average Treewidth	1.33	2.07	1.52
Maximal Treewidth	3	6	5
Average Edge Density	1.02	1.18	1.09
$\%_n$ Reentrant	33.4	15.7	19.6
$\%_g$ Cyclic	0.0	29.9	0.3

Table 1: Some graph statistics (validation data).

Complexity in parsing In syntactic parsing, and to some degree also in semantic parsing, it has long been established that longer sentences are more difficult to parse (McDonald and Nivre, 2011; Van Noord et al., 2018). This is commonly attributed to increasing probability of linguistically more complex structures, as well as to error propagation. However, in the semantic parsing and meaning representation sphere, there is (to the best of our knowledge) little research into what the possible causes of this behaviour are, and whether it is a universal phenomenon across frameworks. Therefore, what we observe with EDS in Figure 4 is more surprising than, intuitively, the PTG and AMR scores, and raises questions about what causes these differences. Hypothetically, either the PERIN parser could be particularly tuned to parse into EDS with great accuracy (which is, for all we know, not the case), or there is a specific property of the EDS framework that differentiates it from the other two frameworks in parser performance related to structural graph complexity.

4. Graph Statistics

In the previous section, we analysed parser performance across three frameworks. Our findings raised questions about the difference between expected and observed behaviour with regards to graph complexity. We now provide a more detailed analysis of the underlying properties of the three frameworks.

We begin by presenting an analysis of struc-

Framework	All	01	02	03	04	05	06	07	08	09	10
EDS	*33.4*	31.15	33.17	33.64	34.01	33.78	33.73	33.55	33.23	33.46	33.39
PTG	*15.7*	3.65	8.25	11.61	13.41	15.39	15.09	16.42	17.67	17.60	21.52
AMR^{-1}	*19.6*	0.19	9.65	16.93	16.55	19.29	20.33	20.59	20.95	20.78	21.35

Table 2: Per-framework percentages of reentrant nodes broken down by graph size decile bins.

tural graph statistics, according to Kuhlmann and Oepen (2016). Table 1 shows a subset of properties computed by the mtool graph analyser. EDS graphs, on average, have the largest number of nodes, with AMR graphs being substantially smaller; in terms of the number of distinct edge labels, the order is reversed. The next four rows in Table 1 seek to quantify degrees of "treeness". Unlike in EDS, about one quarter of PTG and AMR graphs actually are rooted trees. Conversely, the EDS graphs have lower average and maximal treewidth, with PSG appearing least "treelike" in this perspective. The average number of edges per node and percentage of reentrant nodes indicate that EDS has comparatively low edge density but that reentrancies nevertheless occur in one of three nodes, compared to around 16% and 20% for PTG and AMR, respectively. Finally, EDS and AMR exclude cyclic graphs by design, whereas cycles are both allowed and common in PTG.

From the graph structure properties in Table 1, a noteworthy difference between EDS and the other two frameworks is the frequency of reentrant nodes. As discussed previously, reentrancies are central properties that distinguish graph-based structures from tree-based structures and make them more challenging to parse into (Szubert et al., 2020). From previous research, one might assume that this makes reentrant nodes harder to predict, so this datapoint is, again, somewhat surprising.

In Table 2 we look further into the "$\%_n$" row in Table 1 and break the statistics down by complexity bins. We find that the percentage of reentrant nodes is consistently high across bins in EDS, while the percentage of reentrant nodes grows with graph complexity for PTG and AMR (in correlation with a drop in performance). EDS still remains the outlier, while based on observations on PTG and AMR, it could be hypothesised that reentrancies are harder to predict.

To explore this hypothesis, we further refine the methodology used in Section 3, and compare structural parser performance considering the reentrant status of edges in the graphs. The results are charted in Figure 5. The leftmost column shows parser performance considering only edges that are not part of a reentrancy, i.e. do not point to a reentrant node. The rightmost column shows parser performance considering only reentrant edges, i.e. edges that point to a reentrant node. To facilitate comparison, the middle column repeats the data shown in Figure 4, showing performance on all edges.

We observe that EDS performance drops by nearly ten percentage points when going from scoring all edges to only scoring non-reentrant edges, and furthermore observe a slight improvement when considering reentrant edges only.

In the case of PTG and AMR, while there is little difference in performance overall across the three scoring methods, we do see some improvement in the lower decile bins specifically when scoring reentrant edges only.

Considering that PTG and AMR show no deterioration in performance on average for reentrant edges compared to non-reentrant ones, and that EDS performs much better on reentrant edges overall, it would appear that our initial hypothesis is not confirmed – and arguably even disproven.

This motivates a more detailed analysis of reentrant nodes – their causes and kinds of manifestations in the frameworks, which we present in the following section.

5. Reentrancy

The description papers and annotation guidelines for each of the three frameworks in focus mention reentrant nodes to varying degrees, but unlike discussions of reentrancies in AMR in work such as Szubert et al. (2020); Van Noord and Bos (2017), there is (to the best of our knowledge) little in-depth discussion of linguistic phenomena that give rise to reentrancies, or the structures in which they manifest, for EDS and PTG.

In this section we investigate the most frequent and overlapping causes for reentrancies in all three frameworks, with the hopes of learning more about the difficulties – or advantages – of parsing into reentrancies.

To illustrate our approach, Figure 6 shows the EDS graph for the example sentence

(1) *The high interest rates and outlooks announced today surprised and shocked investors.*

This example exhibits some interesting linguistic

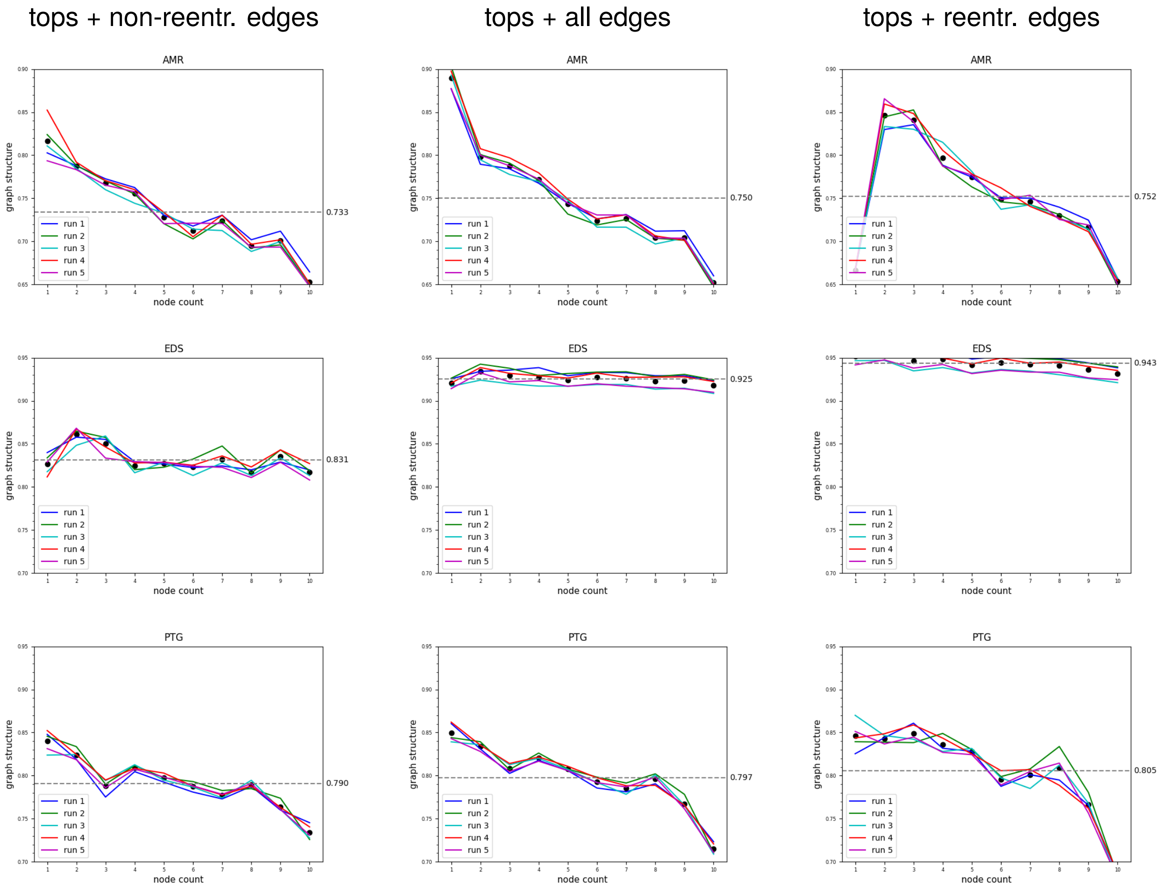


Figure 5: Graph structure F-score, scoring tops and (left) all edges except incoming reentrancies; (center) all edges; (right) only incoming reentrancies, over five train-and-test runs of the PERIN parser.

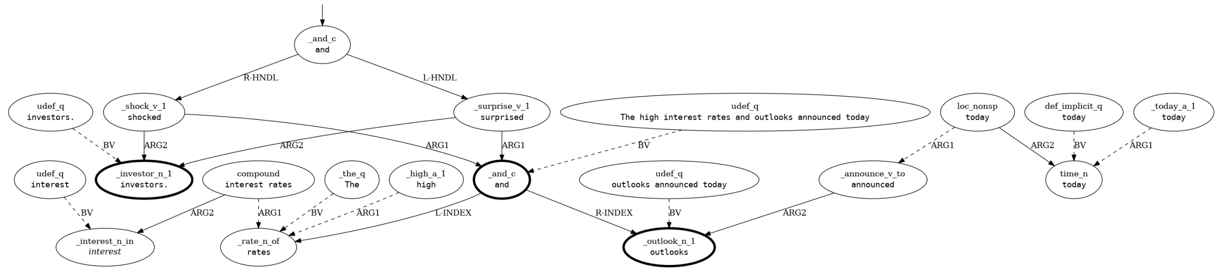


Figure 6: EDS graph for Example (1). Ten automatically annotated reentrant edges are shown as dashed arrows; three remaining reentrant nodes are highlighted with bold edges. In addition to EDS-specific node labels, each node indicates (in typewriter font) the corresponding sub-string of the example.

complexity, including a nominal compound, nominal and verbal coordination, where in the latter both the subject and an extracted object are shared arguments between the conjuncts, a reduced relative clause, and a semantically decomposed temporal modifier. Argument sharing in coordinate structures and relative clauses gives rise to reentrant nodes in the graph, both in EDS and in the other frameworks in our study. Additionally, restrictive modification typically causes reentrancies in EDS, e.g. the attributive adjective, analysis as the compound structure parallel to an unexpressed preposition, and the attachment and internal structure of the temporal modifier. If translated to a more conventional logical-form representation, this would correspond to something like $\texttt{\_high\_a\_1}(x) \wedge \texttt{\_rate\_n\_of}(x)$. Similar reentrancies related to modifier structures will arise in AMR, though not in PTG, where modifiers tend to be dependents of the nodes they modify. Finally, among our three frameworks, EDS has the unique property of encoding quantificational structure, using

Framework	Type	Freq.
EDS	quantification	.377
	compound	.062
	modification	.070
	numeric	.022
	preposition	.072
	other modification	.027
PTG	paratactic structures	.341
AMR	modification	.417

Table 3: Relative frequencies of reentrancy types labelled automatically.)

designated BV ("bound variable") edges. This applies to both determiners that introduce quantificational force (_the_q in our example) and to covert (unexpressed) quantificational predicates, e.g. on bare nominals and in the decomposition of *today* (udef_q and def_implicit_q). These edges, again, reflect the underlying logical structure and are a very frequent source of reentrancies in EDS which is not present in the other frameworks.

5.1. Pilot Annotation

Based on the methodology of Szubert et al. (2020), we begin by empirically observing reentrancies in the frameworks, and assign labels based on the linguistic phenomena they embody. With the goal of quantifying our findings, we carry out a small-scale pilot annotation study on sample sets from each of the three frameworks. For each framework, we build a sample of 150 sentence graphs, randomly selected to include at least one reentrant node, and balanced proportionally across the decile bins.

5.1.1. Predictable reentrancies

From our initial observations of the samples, we find a number of frequent and predictable reentrancy patterns in each of the frameworks that take up a not inconsiderable portion of the data and human labour during annotation. We automate the annotation of these "predictable" reentrancies, and focus manual annotation efforts on the remaining reentrant nodes. Table 3 lists these predictable reentrancies and their relative frequencies in the sample sets.

In the case of EDS, the largest portion of these is taken up by determiners and other quantificational nodes, denoted with the BV edge label (in further discussion, we label these as "category 1" reentrancies). Apart from being a very frequent cause of reentrancies, this edge type is a framework idiosyncrasy and, thus, not a comparable linguistic feature across frameworks. Similarly, we automatically annotate compounds, adjectival modifiers, cardinal and ordinal number, prepositions, and other predictable instances of what EDS analyses as restrictive modification (appositions, possessives), all of which we consider "category 2" reentrancies.

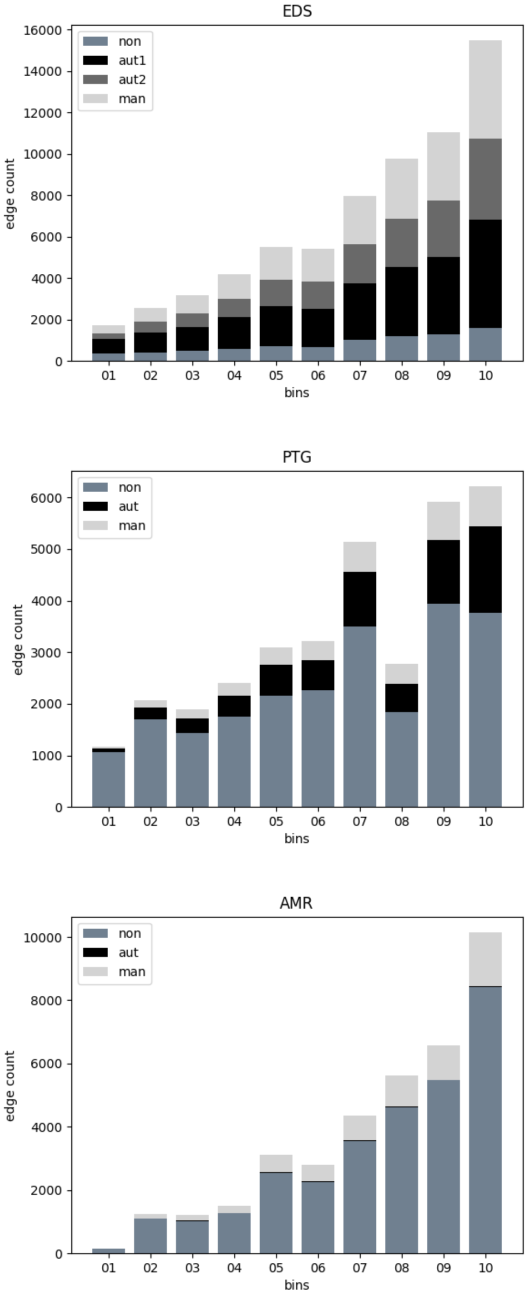


Figure 7: Proportions of edges not involving reentrancy vs. automatically and manually annotated reentrant ones

In the case of PTG, the majority of predictable reentrancies is caused by the framework formality of introducing member and effective edges for all paratactic clause-like structures, be they clauses or compounds, predominantly involving coordination.

Finally, in the case of AMR, a relatively frequent and predictable cause of reentrancy is restrictive modification, as uniquely denoted by the domain edge label.

Figure 7 charts proportions of edges in the full dataset, by framework and complexity, according reentrancy status: not reentrant, part of an automatically annotatable reentrancy (categories 1 and 2 for EDS), or part of a reentrancy requiring manual annotation. These figures give a sense of the portion of reentrancies caused by "predictable" framework formalities like quantification or paratactic structures in EDS and PTG, respectively.

5.1.2. Manual annotation

Following the approach of Szubert et al. (2020), we empirically observe the occurrence of reentrancies and note their causes, starting with the relation set discussed in the original paper, and expanding with more reentrancy type labels as needed. The results of the manual annotation are presented in Table 4, highlighting the most frequent reentrancy phenomena for each of the frameworks.

- Characteristic of EDS, but not captured by the modification-labelling step of the automatic annotation, *comparative* comprises edges incoming from nodes denoting comparative and superlative modifications of adjectives and verbs, as in the sentence fragment *"the largest and most prized market"*.

- The *Control structures* label encompasses various types of argument sharing found in control structures, such as subject and object control, adjunct control, etc., including nominal control.

- Both *coordination* and *coreference* may give rise to argument sharing, and hence reentrancies, as in the example sentence *"the trust said it has rebuilt reserves and improved operations"*.

- The *modal* label covers all reentrancies occurring as a result of modal verb structures, such as *"they may rise to mountainous proportions"*, where the subject is the argument of both the modal verb and the main verb (in the case of PTG), or the main verb gets an additional incoming edge from the modal verb (in the case of EDS and AMR).

- In EDS and PTG, *modification* includes reentrancies occurring from adjectival participles in restrictive modification, as in *"We make waves under controlled conditions and learn where there are buried rock structures."*

- In PTG, *named entities* and similar compound structures of proper nouns also give rise to reentrancies, by linking each constituent to the predicate node, and each other via a *named entity* edge label, such as in the example sentence *"Goldman, Sachs & Co. will manage the offering."*

- In AMR, *partitive* encompasses a wide range of part-of relations that cause reentrancies, as the example of *finger* and *king* in *"I am more powerful than the finger of a king."*

- *Possessive* relations give rise to reentrancies in all three frameworks, by linking the possessor and the object of possession, as in the sentence fragment *"with regard to man's life in society"*.

- For all three frameworks, the *relative clause* is a common cause of reentrancy, with multiple incoming nodes for the shared argument, as in the fragment *"a tile bridge spanning a stream that flows into the building from outside"*.

- Most frequent in AMR, reentrancies labelled *verbalisation* arise from the annotation convention of maximising the use of predicates. This most often manifests as adjectival participles, or nouns as in the example of the (*govern-01, organization*) node pair representing the noun *government*.

- Finally, the *other* label comprises other causes for reentrancies that had less than five occurrences in the sample data for all three frameworks, such as object raising or various discourse elements.

Since data for the three frameworks are not drawn from the same source, the relative frequencies in Table 4 are not horizontally comparable. However, within the frameworks, certain highlights emerge. For example, both EDS and AMR reentrancies prominently feature verbalisation, particularly adjectival participles. Regardless of the different source material, coreference is a frequent cause of reentrancy in both PTG and AMR, while the highly reentrant EDS has an arguably more balanced occurrence of many of the discussed reentrancy types.

Alongside the relative frequencies of reentrancy types in the sample data, Table 4 shows the error rates of the PERIN parser for the respective reentrancy types, bringing us back to the original question of parser performance and where we might see particular areas of improvement.

For example, in the case of coreference in PTG and AMR, the PERIN parser fails to produce the correct graph structure 38% and 43% of the time,

	EDS		**PTG**		**AMR**	
Type	Freq.	Error	Freq.	Error	Freq.	Error
clause-like structures	.067	.058	-	-	.039	.428
comparative	.027	.000	-	-	-	-
control structures	.063	.250	**.136**	.291	**.106**	.342
coordination	**.127**	.125	**.165**	.258	**.134**	.166
coreference	.027	.142	**.401**	.382	**.243**	.436
modal	.011	.333	.014	.400	.008	.333
modification	**.167**	.071	**.176**	.451	-	-
named entity	-	-	.031	.818	-	-
partitive	-	-	-	-	.058	.285
possessive	.039	.500	.008	.666	.008	.666
relative clause	**.183**	.065	.022	.125	.008	.666
verbalisation	**.159**	.075	.002	.000	**.326**	.358
other	.122	.173	.039	.571	.067	.541

Table 4: Relative frequencies and error rates of manually annotated reentrancy types.

Framework	**EDS**	**PTG**	**AMR**
Missed/Total	.106	.345	.405

Table 5: Ratio of reentrant edges the parser failed to produce vs. total number of reentrant edges, per framework (sample set).

respectively, implying that, given the prominence of coreference in the frameworks, correct coreference resolution has an impact on parser performance, especially in longer sentences with more occurrences of this reentrancy structure.

Similarly, with verbalisation being a frequent cause of reentrancies in AMR, the parser demonstrates a 35% error rate on this reentrancy type. As with the previous example, better performance on this task would likely significantly increase parser performance overall.

In the case of possessives, it is interesting to note that, although this particular reentrancy cause makes up a relatively small proportion of reentrancies observed in the annotation set, the parser shows an error rate of 50% or greater for possessives in all three frameworks.

Table 5 summarises the error rates over the total number of reentrant edges per framework, in the sample annotation set. Note that this view does not include the "predictable" reentrancies from the automatic annotation step. Even disregarding these frequent reentrancy types that are a result of framework-specific regularities and, therefore, may be somewhat easier for the parser to correctly predict, the parser retains the lowest parsing error rate on EDS, with just 11% of reentrant edges not produced. Unlike for PTG and AMR, EDS annotations were guided by a large-scale computational grammar, i.e. automatically confirmed to obey formal principles of derivation and composition, which may both lead to higher degrees of predictability and overal greater consistency of the annotations.

6. Conclusion

Building on previous research into parser performance for different frameworks of meaning representation graphs, we carried out a contrastive study focussing on patterns of parser behaviour in sentences of increasing graph complexity, and the presence and frequency of reentrant nodes in the target graph. We performed a small-scale, semi-automated annotation effort over a sample of our datasets, and discussed observations on common linguistic phenomena that give rise to reentrant structures, and how successful a state-of-the-art parser is in producing them. This work sets the foundation for future focussed parser development, as well as further discussions of the particularities of framework design and annotation guidelines.

We intend to explore both of these tracks in future work, specifically to reach out to the PERIN developers and discuss properties of the parsing architecture that may explain our findings (and potential revisions to mitigate their negative impact on parser performance). In the framework analysis track, we will explore redefining graph complexity (and, subsequently, binning) by reentrancy count, in contrast to the currently used node-count approach. We also intend to refine and scale up the reentrancy annotation effort, to produce a larger dataset with phenomena categories aligned across frameworks, and to the highest degree possible over the same strings.

7. Acknowledgements

We would like to thank the reviewers for their insightful comments and suggestions regarding improvements to the paper and future work.

8. Bibliographical References

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 *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*, pages 178–186.

Maja Buljan, Joakim Nivre, Stephan Oepen, and Lilja Ovrelid. 2020. A tale of three parsers: Towards diagnostic evaluation for meaning representation parsing. In *12th International Conference on Language Resources and Evaluation (LREC), MAY 11-16, 2020, Marseille, FRANCE*, pages 1902–1909. European Language Resources Association (ELRA).

Maja Buljan, Joakim Nivre, Stephan Oepen, and Lilja Øvrelid. 2022. A tale of four parsers: methodological reflections on diagnostic evaluation and in-depth error analysis for meaning representation parsing. *Language Resources and Evaluation*, 56(4):1075–1102.

Shu Cai and Kevin Knight. 2013. 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.

Ann Copestake, Dan Flickinger, Carl Pollard, and Ivan A. Sag. 2005. Minimal Recursion Semantics. An introduction. *Research on Language and Computation*, 3(4):281 – 332.

Dan Flickinger, Stephan Oepen, and Emily M. Bender. 2017. Sustainable development and refinement of complex linguistic annotations at scale. In Nacy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*, pages 353 – 377. Springer, Dordrecht, The Netherlands.

Jan Hajič, Eva Hajičová, Jarmila Panevová, Petr Sgall, Ondřej Bojar, Silvie Cinková, Eva Fučíková, Marie Mikulová, Petr Pajas, Jan Popelka, Jiří Semecký, Jana Šindlerová, Jan Štěpánek, Josef Toman, Zdeňka Urešová, and Zdeněk Žabokrtský. 2012. Announcing Prague Czech-English Dependency Treebank 2.0. In *Proceedings of the 8th International Conference on Language Resources and Evaluation*, pages 3153 – 3160, Istanbul, Turkey.

Marco Kuhlmann and Stephan Oepen. 2016. Towards a catalogue of linguistic graph banks. *Computational Linguistics*, 42(4):819–827.

Artur Kulmizev, Miryam de Lhoneux, Johannes Gontrum, Elena Fano, and Joakim Nivre. 2019. Deep contextualized word embeddings in transition-based and graph-based dependency parsing–a tale of two parsers revisited. *arXiv preprint arXiv:1908.07397*.

Ryan McDonald and Joakim Nivre. 2011. Analyzing and integrating dependency parsers. *Computational Linguistics*, 37(1):197–230.

Stephan Oepen, Omri Abend, Lasha Abzianidze, Johan Bos, Jan Hajic, Daniel Hershcovich, Bin Li, Tim O'Gorman, Nianwen Xue, and Daniel Zeman. 2020. MRP 2020: The second shared task on cross-framework and cross-lingual meaning representation parsing. In *Proceedings of the CoNLL 2020 Shared Task: Cross-Framework Meaning Representation Parsing*, pages 1–22, Online. Association for Computational Linguistics.

Stephan Oepen, Omri Abend, Jan Hajic, Daniel Hershcovich, Marco Kuhlmann, Tim O'Gorman, Nianwen Xue, Jayeol Chun, Milan Straka, and Zdenka Uresova. 2019. Mrp 2019: Cross-framework meaning representation parsing. In *Proceedings of the Shared Task on Cross-Framework Meaning Representation Parsing at the 2019 Conference on Natural Language Learning*, pages 1–27.

Stephan Oepen and Jan Tore Lønning. 2006. Discriminant-based mrs banking. In *LREC*, pages 1250–1255.

Juri Opitz. 2023. Smatch++: Standardized and extended evaluation of semantic graphs. *arXiv preprint arXiv:2305.06993*.

David Samuel and Milan Straka. 2020. UFAL at MRP 2020: Permutation-invariant semantic parsing in perin. *arXiv preprint arXiv:2011.00758*.

Petr Sgall, Eva Hajičová, and Jarmila Panevová. 1986. *The Meaning of the Sentence and Its Semantic and Pragmatic Aspects*. D. Reidel Publishing Company, Dordrecht, The Netherlands.

Ida Szubert, Marco Damonte, Shay B Cohen, and Mark Steedman. 2020. The role of reentrancies in abstract meaning representation parsing. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2198–2207.

Rik Van Noord, Lasha Abzianidze, Antonio Toral, and Johan Bos. 2018. Exploring neural methods for parsing discourse representation structures. *Transactions of the Association for Computational Linguistics*, 6:619–633.

Rik Van Noord and Johannes Bos. 2017. Dealing with co-reference in neural semantic parsing. In *Proceedings of the 2nd Workshop on Semantic Deep Learning (SemDeep-2)*, pages 49–57. Association for Computational Linguistics (ACL).

Daniel Zeman and Jan Hajic. 2020. FGD at MRP 2020: prague tectogrammatical graphs. In *Proceedings of the CoNLL 2020 Shared Task: Cross-Framework Meaning Representation Parsing*, pages 33–39.

Mapping Czech Verbal Valency to PropBank Argument Labels

Jan Hajič, Eva Fučíková, Markéta Lopatková, Zdeňka Urešová
Charles University, Faculty of Mathematics and Physics, Institute of Formal and Applied Linguistics
Malostranské nám. 2/25, 118 00 Prague 1, Czechia
{hajic,fucikova,lopatkova,uresova}@ufal.mff.cuni.cz

Abstract

For many years, there have been attempts to compare predicate-argument labeling schemas between formalisms, typically under the dependency assumptions (even if the annotation by these schemas could have been performed on constituent-based specifications). Given the growing number of resources that link various lexical resources to one another and thanks to parallel annotated corpora (with or without annotation), it is now possible to do more in-depth studies of those correspondences. We present here a high-coverage pilot study of mapping the labeling system used in PropBank (for English) to Czech, which has so far used mainly valency lexicons (in several closely related forms) for annotation projects, under different levels of specification and different theoretical assumptions. The purpose of this study is both theoretical (comparing the argument labeling schemes) and practical (to be able to annotate Czech under the standard UMR specifications).

Keywords: predicate-argument structure, valency, syntax, semantic, semantic roles, PropBank, Prague Dependency Treebank, SynSemClass, Unified Verb Index

1. Introduction

PropBank (Kingsbury and Palmer, 2002; Palmer et al., 2005), as it is usually referred to, is an English treebank (usually meant to be the Penn Treebank, Marcus et al., 1993, in its many [later] versions) annotated with a predicate-argument structure, and in addition, with semantic roles, as defined in (Palmer et al., 2005). The individual verbs (further sub-divided into verb senses, and assembled in so-called “PropBank Frame Files”), form a lexicon containing proper set of argument labels for each of their senses. The corpus and the lexicon (frame files), with the annotation guidelines as the main source of the underlying theoretical description, form a specification of a fundamental view of the predicate-argument structure as applied to English. The success of such predicate-argument annotation has led to the creation of several treebanks in other languages annotated in the Penn-Treebank-style, and “propbanked” using the specification for English, such as Arabic, Basque, Chinese, Finnish, Hindi, Persian, Portuguese, Turkish and Urdu, as well as the multilingual collection of the IBM Universal Proposition Banks for 23 languages (Jindal et al., 2022).[1] The PropBank scheme of labeling predicate-arguments has been also used for the Abstract Meaning Representation (AMR) annotation (Banarescu et al., 2013) (with some specific predicates added) and recently, also for the Uniform Meaning Representation (UMR) annotation (Van Gysel et al., 2021; Wein and Bonn, 2023) (with even more abstract predicates added). Both AMR and UMR guidelines[2] call, in principle, for the same predicate-argument labeling scheme as in the original PropBank.

The Czech language valency scheme, essentially also a predicate-argument labeling scheme, is however based on the Functional Generative Description (dependency) theory (Sgall et al., 1986), which treats especially the first two verb arguments differently than PropBank and uses a different specification and labeling style for the remaining arguments. It is used in the main Czech valency dictionaries (Urešová et al., 2014; Lopatková et al., 2022): VALLEX (Žabokrtský and Lopatková, 2007; Lopatková et al., 2016) and PDT-Vallex (Hajič et al., 2003). The latter has been used in the Prague Dependency TreeBank (PDT) in the annotation of the four PDT-C (Hajič et al., 2020) subcorpora annotated on the so-called Tectogrammatical Layer, or Tectogrammatical Representation (TR), which is “deeper” than the traditional dependency syntax used in the Analytical Layer (surface syntax) of the PDT(-C) (Hajič et al., 2020) or in the Universal Dependencies annotation scheme.

At the same time, lexical semantic resources[3] have been increasingly available in an interlinked form. That covers both linking across such lexicons, and/or linking them across languages. An example

[1] The UP 2.0 project creates the resulting annotated files, at least for some languages, by an automatic conversion from the UD-style annotation. This includes all the Czech UD treebanks as well, which means that the resulting labeling depends almost purely on the UD syntactic scheme.

[2] https://umr4nlp.github.io/web/guidelines.html

[3] We are interested primarily in verbal lexical resources, but other resources are being linked together too, e.g., in the Linguistic Linked Open Data project https://pret-a-llod.github.io.

of such linking[4] is the Unified Verb Index[5] (Palmer et al., 2014; Stowe et al., 2021) and the multilingual SynSemClass ontology and lexicon[6] (Uresova et al., 2020), which has a rich set of links to PropBank, FrameNet, VerbNet, WordNet for English, and to the VALLEX and PDT-Vallex lexicons for Czech. In addition, the CzEngVallex lexicon[7] (Urešová et al., 2015a) links Czech and English verb entries,[8] using the PDT scheme for Czech. It is also important to note that the EngVallex lexicon, used as a basis for the bilingual CzEngVallex, was built upon PropBank - albeit it also uses the PDT argument labeling scheme - and contains (some) links back to the original PropBank frame files (Cinková, 2006).

The goal of this paper is to describe a recent attempt at a large-scale, large-coverage mapping of the predicate-argument labeling schemas: the PDT-based valency approach and the PropBank approach, applied to Czech. Mapping means to try to capture the same predicate-argument relations (as found in the Czech valency dictionaries) using the PropBank specifications (by mapping the labels of predicate-argument relations). The results will help to see the theoretical differences, and will perhaps also lead to an easier annotation of Czech within the UMR scheme (which also uses the PropBank argument labeling).[9] While there are several theoretical questions to answer, there are also more practical issues and open questions (and benefits if the differences can be explicitly and formally described):

- How is the PropBank approach different from the semantic point of view, especially in the labeling of the first two arguments?
- Can an algorithm be designed to convert, for a particular verb sense, its PDT-based valency structure into the PropBank predicate-argument labeling scheme?
- If yes, what are the biggest differences that cause complications or lead to the impossibility of mapping to the PropBank scheme exactly?
- What information from the richly annotated lexical resources, such as SynSemClass and PropBank, and the associated bilingual corpora between Czech and English can be used?

2. Related Work

Mapping the (English) PropBank scheme to other languages has been researched previously. The PropBanks mentioned in the Introduction have used some form of mapping. For example, Xue et al. (2002) describes a mapping for Chinese. The first comparative study on English and Czech valency draws a comparison between PropBank, LCS Database, and PDT (Hajičová and Kučerová, 2002). Further, for English, the relations between the PropBank arguments and the valency slots as defined in the PDT scheme have been described by Cinková (2006). The resulting EngVallex lexicon has then been used for the tectogrammatical annotation of English in the Prague Czech-English parallel Dependency Treebank (PCEDT,[10] Hajič et al., 2012).

Studies on English-Czech valency using treebank examples or treebank token alignment are described in (Šindlerová and Bojar, 2009; Bojar and Šindlerová, 2010) and resulted in the creation of a bilingual Czech-English valency lexicon - CzEngVallex - described in (Urešová et al., 2015b, 2016). Detailed studies on aligning English and Czech arguments also exist, such as (Šindlerová et al., 2015). However, all these studies compare the valency solely under the PDT labeling scheme.

A comprehensive description comparing Czech PDT-based valency and the English PropBank labeling schema is presented in the papers (Urešová et al., 2014) and (Xue et al., 2014). They provide a detailed inspection of argument labeling differences between Czech and English annotation within the AMR scheme. As the study (Urešová et al., 2014) reveals, the by far most frequent mismatch is caused by different argument labeling. While there is a complete match for most purely transitive verbs, there is a discrepancy for most other verbs since PropBank continues to number arguments of corresponding verbs consecutively but PDT-Vallex attempts the semanticization of argument labels: `ADDR` (addressee), `EFF` (effect) and `ORIG` (origin). These two studies have been made on a very small subset of verb frames: Xue et al. (2014) use only 100 sentences and verbs found in them.

Finally, a detailed study of mappings between the structures used in AMR and those used in UMR are presented in (Bonn et al., 2023). However, here the Czech AMR annotation uses the Czech PDT-Vallex valency lexicon labels, while the English AMR uses the standard English PropBank Roleset Lexicon (Frame Files).

[4] Among others, such as BabelNet (Navigli and Ponzetto, 2012), Predicate Matrix (Lopez de Lacalle et al., 2016), LLOD etc.

[5] https://uvi.colorado.edu

[6] https://lindat.mff.cuni.cz/services/SynSemClass50, http://hdl.handle.net/11234/1-5230

[7] http://hdl.handle.net/11234/1-1512

[8] https://lindat.mff.cuni.cz/services/CzEngVallex

[9] While UMR does not strictly require the PropBank approach, it is understood that having a unified argument labeling scheme is an advantage.

[10] http://hdl.handle.net/11858/00-097C-0000-0015-8DAF-4

3. Data Sources

The datasets used for pre-assigning the PropBank-defined arguments to the PDT-based valency frames and their individual slots have been the following:

- **PropBank Frame Files** taken from the current github version of PropBank;[11] see Sect. 3.1,
- **CzEngVallex bilingual valency lexicon** available in the LINDAT/CLARIAH-CZ repository[12] (Urešová et al., 2016), see Sect. 3.2,
- **SynSemClass ontology 5.0**[13] (Urešová et al., 2023), see Sect. 3.3.

In the following sections, we will present the basic structure of these resources stressing the predicate-argument labeling scheme and properties.

3.1. PropBank and PropBank Frame Files scheme

The original Proposition Bank project (Palmer et al., 2005) "took a practical approach to semantic representation, adding a layer of predicate-argument information, or semantic role labels, to the syntactic structures of the Penn Treebank" (Marcus et al., 1999). In fact, one of the original motivations was to define **semantic roles** for the annotation for each verb used in the corpus, with the alterations appearing in the corpus being one of the important points. It was clearly stated that syntax alone is not sufficient to generalize (or, better to say, uniformly annotate) over various forms of expressions to represent the same meaning in relation to the verb arguments. This approach can be demonstrated on the verb *break* appearing in two syntactically distinguished constructions: *John broke the window.* and *The window broke.* In both cases, the affected object is the window, syntactically expressed as an Object in one case and Subject in the other. There are many verbs behaving similarly, such as *play* (*The sergeant played taps.* vs. *Taps played quietly in the background.*)[14] or *load* (*He loaded the truck with hay.* vs. *He loaded hay onto the truck.*).

As a result, PropBank uses an approach (at the top-level abstraction) similar to that of the PDT (Sect. 3.2), i.e., using a list of arguments specific for each verb. However (as opposed to the PDT), PropBank, while using numbered argument roles, defines Arg0 for a prototypical Agent and Arg1 for a prototypical Patient (or Theme), following (Dowty, 1991). I.e., in the aforementioned example of the two uses of *break*, the *window* argument will always be marked as Arg1 to signal the same semantic "position" relative to the verb *break*, regardless of the syntactic structure; as a consequence, in the case of the second example sentence, the verb *break* will have no argument labeled Arg0. Furthermore, each sense of the verb lemma has a separate **roleset** (denoted by an ordinal number attached to the lemma, such as *kick.02*), and they are collected in one frame file for a given lemma.

The original definition of a roleset in the frame files required a description associated with each argument, such as "sayer" for Arg0 of the verb (sense) *suggest.01* or "utterance (suggestion)" for its Arg1, or "chart-maker" for Arg0 of *chart.01* or "thing being charted" for its Arg1.[15] However, these descriptions are not formally defined, so they are unique for each roleset, and not related (much) to the same description at a different roleset.[16] Also, they do not generalize over "content" synonymy (as in *buy* and *sell*, as the original FrameNet did by putting them to a single frame labeled COMMERCE)[17] - the description of Arg0 for *sell* is "seller" while the same description is used for Arg2 of *buy*. Similarly, PropBank does not group what would be called synonyms, e.g., in WordNet (Fellbaum, 1998) - it keeps each lemma (and word sense) separately. However, thanks to mappings from PropBank to VerbNet (Schuler and Palmer, 2005), available in PropBank v3.4[18] or in the UVI index,[19] at least the broadly defined semantic classes as represented in VerbNet can be determined.

3.2. CzEngVallex: Parallel Czech-English Valency Lexicon and the PDT Valency Scheme

CzEngVallex (also CEV) is a bilingual Czech-English verbal valency lexicon (Urešová et al., 2015). It includes 20,835 aligned valency frame pairs[20] and their aligned arguments. This lexicon uses data from the PCEDT corpus and also takes advantage of the existing valency lexicons for both

[11] http://propbank.github.io/v3.4.0/frames/index.html

[12] http://hdl.handle.net/11234/1-1512

[13] http://hdl.handle.net/11234/1-5230

[14] Examples from the (Palmer et al., 2005) paper.

[15] https://github.com/propbank/propbank-frames/blob/main/frames/chart.xml

[16] This is similar to the approach of FrameNet, which also declares that a semantic role defined or used in two different frames should not be taken to mean the same. See SynSemClass (Sect. 3.3) for a different approach.

[17] Currently, FrameNet v2 uses two separate frames, Commerce_buy and Commerce_sell, corresponding to the PropBank approach.

[18] http://propbank.github.io/v3.4.0/frames/index.html

[19] https://uvi.colorado.edu/uvi_search

[20] Each valency frame in the PDT-based valency approach essentially corresponds to one verb sense, therefore, the term "verb sense" and the term "valency frame" are used interchangeably (simplifying the matter somewhat given that there are some cases where the difference matters).

languages (PDT-Vallex and EngVallex).

FGD valency theory. The PDT-Vallex and EngVallex lexicons, and subsequently the CzEngVallex, are built upon the valency theory developed within the Functional Generative Description approach (FGD). As described in detail in (Urešová et al., 2016; Lopatková et al., 2016), in this dependency approach, valency is seen as a property of (some) lexical items, mainly the verb being the core of the sentence, to select for certain complementations in order to form larger units of meaning (sentence, phrase, etc.). The valency characteristics (i.e., the number or arguments and morphosyntactic surface realization of the selected dependent elements constituting the valency structure) are represented in the form of (PDT-)valency frames; these frames are listed in valency lexicons.

The basic characteristics of the FGD valency theory can be found in (Panevová, 1994): it combines the syntactic and semantic approach for distinguishing valency elements. The relation between the governor (primarily verb) and its dependent is characterized by so-called *functors* at the tectogrammatical layer: a functor is a label representing the semantic values of a syntactic dependency relation.[21] There are two axes of classifying the valency modifications in the FGD valency theory: the first axis distinguishes inner participants (arguments) and free modifications (adjuncts), and the other axis distinguishes between obligatory and optional complementations.

There are five "inner participants" (arguments): Actor/Bearer (functor ACT), Patient (PAT), Addressee (ADDR), Origin (ORIG) and Effect (EFF). Out of the five argument types, FGD states that the first two are connected with no specific semantics, contrary to the remaining three ones. The first argument is always the Actor (ACT), the second one is always the Patient (PAT). The Addressee (ADDR) is the semantic counterpart of an indirect object that serves as a recipient or simply an "addressee" of the event described by the verb. Effect (EFF) is the semantic counterpart of the second indirect object describing typically the result of the event (or the contents of an indirect speech, for example, or a state as described by a verbal attribute – the complement). Origin (ORIG) also comes as the second (or third or fourth) indirect object, describing, not surprisingly, the origin of the event (in the "creation" sense, such as to build from metal sheets, not in the directional sense).

FGD valency theory has further adopted the concept of shifting of "cognitive roles". According to this special rule, semantic Effect, semantic Addressee and/or semantic Origin are being shifted to the Patient (PAT) position in case the verb has only two arguments.[22]

In addition to the inner participants, FGD distinguishes about 50 types of semantically determined adjuncts (free modifications), such as temporal, locative or causal. Due to the "free nature" of adjuncts, only the presence of arguments (obligatory or optional) and obligatory adjuncts is recorded in verbal valency frames.

The FGD-based valency lexicons (PDT-Vallex, EngVallex, and CzEngVallex). CzEngVallex (CEV) has been developed together with the PCEDT corpus[23] (Hajič et al., 2012), i.e., a sentence-parallel treebank based on the sentences of the Wall Street Journal section of Penn treebank[24] and their manual translations. This annotation includes also verb sense annotation by links to valency frames in PDT-Vallex (for Czech) and EngVallex (for English).

PDT-Vallex (Urešová, 2011) has been developed as a resource for valency annotation in the PDT. This lexicon is publicly available as a part of the PDT version 2 published by the Linguistic Data Consortium and also separately.[25] The version of PDT-Vallex used for CzEngVallex contains 11,933 valency frames for 7,121 verbs.

EngVallex (Cinková, 2006) was built within the same FGD valency theory and makes use of PropBank, from which it was automatically preconverted and subsequently manually refined and used for the tectogrammatical annotation of the Wall Street Journal section of the Penn Treebank. EngVallex contains 7,148 valency frames for 4,337 verbs.

3.3. SynSemClass Ontology

SynSemClass (SSC) (Urešová et al., 2023) is an event-type ontology for multiple languages. It includes Czech, English (Urešová et al., 2019a)), German (Urešová et al., 2021) and Spanish words and definitions (Fernández-Alcaina et al., 2022). In SynSemClass, contextually-based synonymous verbs in various languages are classified into event-type concepts, or **multilingual synonym classes**

[21] For a full list of all PDT dependency relations and their labels (functors), see (Mikulová et al., 2005).

[22] This can be illustrated on the sentence *The teacher asked the pupil* where the semantic Addressee (*the pupil*) is shifted to the Patient position and thus gets the PAT functor. This rule, when viewed from the annotation point of view, helps to keep consistency at the expense of lower "semantic adequacy".

[23] http://hdl.handle.net/11858/00-097C-0000-0015-8DAF-4

[24] https://catalog.ldc.upenn.edu/LDC99T42

[25] https://lindat.mff.cuni.cz/repository/xmlui/handle/11858/00-097C-0000-0023-4338-F

according to the semantic and syntactic properties they display. To have empirical evidence for such classification, SynSemClass is being developed in a "bottom-up" fashion: The candidate verbs for synonym classes are taken from actual examples from parallel English-Czech, English-German, or English-Spanish corpora.

As described in detail in (Urešová et al., 2020, 2019b, 2018a,c,b), SSC synonym classes are characterized by the following main features:

- The **name** of each **class** stands for a single concept (e.g., of *eating*)[26] and corresponds to the verb that represents the prototypical sense, in each of the languages included.
- Each class is provided with a brief general **class definition** in each language included, which characterizes the meaning (concept) of the class.
- For each class, SynSemClass also provides a fixed set of "situational participants", labeled with SSC **semantic roles** common for all the class members in that class. The roles are mapped to the predicate-argument (valency) structure of the individual class members. Thus, they are characterized both meaning-wise (semantic roles) and structurally-wise (valency arguments). When mapping the roles from a given set of participants, each one must be realised as "something" taken from the valency frame of a verb in that class.
- Each verb (sense) included in a given SSC synonym class is linked to one or more lexical resources for the given language. In SSC, there are links to e.g., VALLEX[27] for Czech, FrameNet[28] and VerbNet[29] for English, E-VALBU[30] for German, AnCora[31] for Spanish, among others.

  Further, each verb is exemplified by instances of real texts extracted from translated or parallel corpora. Specifically, data is extracted from the Prague Czech-English Dependency Corpus (PCEDT)[32] for Czech-English, from the Paracrawl corpus[33] for German-English and from the X-SRL dataset[34] for Spanish-English.

4. Mapping PDT to PropBank

Given the plethora of richly interlinked resources, as described in Sect. 3, one might wonder why the mapping between arguments of verbs in these resources is ever worth investigating and not just a simple technical problem. The reason is first of all the richness of the language itself and its ambiguity as well as redundancy. In addition, the usual problems related to manually annotated and curated resources are present, too: ambiguous guidelines, (low) inter-annotator agreement, not enough details in lexical resource descriptions, evolving resources over time with changing approaches and annotator teams, and their insufficient coverage (Fučíková et al., 2024).

As an example, let's take the Czech word *sídlit* (lit. *to reside*).[35] It has two core arguments (in the base PDT-Vallex resource): `ACT` for the thing residing somewhere, and `LOC` for the location. In the CzEngVallex resource (Sect. 3.2), the entry for *sídlit* is linked to seven different English verbs (anchor, base, be, ensconce, house, locate, and reside), as collected (and manually filtered and annotated) from the annotated parallel PCEDT corpus. The conflicting argument mappings, when traced from CzEngVallex to PCEDT to PropBank, are shown in Fig. 1.[36]

In the SynSemClass ontology (Sect. 3.3), the Czech verb *sídlit* appears in the class named (in English) `locate`. On top of the aforementioned English equivalents coming from CzEngVallex, there are also some additional English verbs (lie, settle, spread, sit) presumably bearing the same meaning as *sídlit*, with yet another set of mappings traced from the original PDT-like ones to PropBank arguments.

The natural question is whether these mappings can be (automatically) consolidated somehow to serve as a basis for a (manually-based) filtering and editing process to arrive at such a set of PropBank-style arguments for *sídlit* that would respect the PropBank guidelines as much as possible. An algorithm that tries to do exactly that and which makes use of the input resources (as presented in Sect. 3) plus the of parallel Czech-English annotated corpus PCEDT for getting corpus-based preferences, is described in the next section.

[26] This is different from the commonly used term of "semantic classes of verbs" as represented, for example, in VerbNet, where the class is defined much more broadly – such as for all verbs of movement.

[27] https://hdl.handle.net/11234/1-3524

[28] http://framenet.icsi.berkeley.edu/

[29] http://verbs.colorado.edu/verbnet/index.html

[30] https://grammis.ids-mannheim.de/verbvalenz

[31] http://clic.ub.edu/corpus/es/ancoraverb_es

[32] https://ufal.mff.cuni.cz/pcedt2.0/en/index.html

[33] https://paracrawl.eu

[34] https://catalog.ldc.upenn.edu/LDC2021T09

[35] *sídlit* is relatively simple example, given that from the Czech language perspective, there is only one meaning; cf. the old Dictionary of Standard Czech Language, or SSJČ, at https://ssjc.ujc.cas.cz.

[36] The verb "be" has been left out, since it was treated as auxiliary in the corpus.

Figure 1: Source mapping for the arguments of the Czech verb *sídlit* to its English counterparts

sídlit (PDT-based arguments)	reside	anchor	base locate	house ensconce
ACT	→Arg0	→Arg1	→Arg1	→Arg1
LOC	→Arg1	no mapping	→ArgM-LOC	→Arg2

5. The Mapping Algorithm

The automatic mapping is created in two steps, for all Czech verb senses (valency frames) as found in the PDT-Vallex lexicon:

- collecting, for each frame, all possible mappings by tracing the available resources for *each argument separately* to get its possible PropBank argument label(s), together with frequencies of these mappings in available corpora (Sect. 5.1), and
- consolidating and creating the new, complete PropBank-style rolesets for Czech verb senses, with the right number of arguments and their labels (Sect. 5.2).

For cases where the final roleset cannot be determined unambiguously, we collect statistics from the parallel PCEDT corpus,[37] which is annotated by both the Czech and English valency frames and their arguments. These numbers of corpus occurrences are then used in determining the preferred mapping when displaying it to the annotator for making the final decisions.

5.1. Collecting Instances of Argument Mappings from Existing Resources

There are two main sources where the traces leading from the Czech PDT-based valency frame and its individually taken arguments to the PropBank ones come from: CzEngVallex (CEV, Sect. 3.2) and SynSemClass (SSC, Sect. 3.3).

CEV mapping. The mapping(s) of the PDT-style labels (functors), as listed for each PDT-Vallex valency frame, to the PropBank argument labels are collected from CzEngVallex entries, together with the PCEDT corpus frequencies. For example, the Czech verb *asistovat* (one sense only, with two arguments: ACT and PAT) is linked to two single-sense EngVallex entries *assist* and *support* in CEV.[38] From these two entries and their occurrences on the English side of the PCEDT, the following PropBank arguments[39] have been identified (numbers in parentheses indicate occurrences of these mappings in the English part of the PCEDT):

asistovat	assist	support
ACT	→Arg0 (16x)	→Arg0 (102x)
PAT	→Arg1 (20x) →Arg2 (3x)	→Arg1 (149x)

Thus by performing three "hops" - from PDT-Vallex to CzEngVallex to PCEDT to PropBank - we are getting, for *asistovat*, an unambiguous mapping from ACT to Arg0 (attested 118 times in the corpus) and an ambiguous mapping from PAT to both Arg1 (169 times in data) and Arg2 (three times).

SSC mapping. While using CEV gives us technically simple means to arrive at (unambiguous, or ambiguous (frequency-annotated)) mappings to PropBank argument labels, it only covers the PCEDT data. To exploit another highly relevant resource, we are using SSC to collect mappings for more verbs (verb senses/frames) from PDT-Vallex to PropBank argument labels. Instead of using the direct frame-to-frame mappings available in CEV, we use one of the major SSC features, namely the mapping between the semantic roles (common for each multilingual class, and thus shared by the verbal lexical units in several languages, including Czech and English) and the original verb arguments, taken from PDT-Vallex and EngVallex. From these mappings, we can extract direct functor-to-functor mapping (as if from CEV) and consequently the PropBank argument labels based on the links in EngVallex. Given that the SSC classes are much broader than the direct bilingual verbal links in PCEDT, we can get bigger coverage, but also more ambiguity.

Let's start with the SSC class "commit / dopustit_se" in which all verbs (including the English verbs *blunder* and *commit*) share two semantic roles, "Perpetrator" and "Deed". For *blunder*, SSC maps these roles to ACT and PAT, respectively, and

[37] https://ufal.mff.cuni.cz/pcedt2.0/publications/eng_pb_links.txt - actually, only from the English side as the target side of each of the possible mappings, since the Czech frequencies are irrelevant for this task, given we go through all of the verbs found in the lexicon.

[38] ... because *asistovat* had these two translational counterparts in PCEDT.

[39] Please recall that the English side of the PCEDT is in fact the original WSJ portion of the Penn Treebank with PropBank annotation on top of it, see Sect. 3.2.

afterwards EngVallex traces them to PropBank's Arg0 (2 instances) and Arg1 (1×), respectively; for *commit*, "Perpetrator"→ACT and "Deed"→CPHR are traced to Arg0 (9x) and Arg1 (12x), respectively. In this case (since no ambiguity arises), the Args can then be easily mapped to the arguments of all the Czech verbs from the same class, namely *dopustit_se, dopouštět_se, páchat and spáchat*, because we know which of their valency frame functors correspond to "Perpetrator" and which to "Deed".

However, it is common that the resulting mappings are (even highly) ambiguous. Fig. 2 illustrates the case for the Czech verb "líbit se", which is one of the Czech verbs in the class "appeal / líbit_se" (meaning to "like" or "be pleased by" something).

5.2. Mapping PDT Valency Frames to PropBank Rolesets

The final steps are to suggest the mapping for the whole valency frame to the PropBank-style roleset, incorporating the procedure described above for the individual arguments. Since these are the last steps before the manual pass of obtaining a PropBank-style Czech rolesets, we are describing them more technically by referring to the actual worksheet (table)[40] that will be used by the annotators.

Mappings for individual functors. Each verb record consists of several rows – one identifying the verb sense (roleset) and then one for each (original) functor / PDT argument, followed by an empty row. A description of how the rows and columns are filled is described below.

1. For each verb sense (frame) from PDT-Vallex, create its PropBank ID (column A, UMR ID).
   Example: *spolknout* "swallow; eat_up": PDT-Vallex spolknout (v-w6385f1) –> *spolknout-001*.

2. Copy the PDT-Vallex ID and its frame members (functors) to column B (PDT frame), with the verb lemma and frame ID on the same line as the PropBank ID, and the argument functors immediately under that.
   Example: *spolknout-001* "swallow; eat_up" gets link to the PDT-Vallex spolknout (v-w6385f1), its two functors are indicated in separate lines, ACT (in nominative) and PAT (in accusative).

3. If the verb sense occurs in some SSC class(es), put its class ID and its semantic roles to the appropriate rows corresponding to the role-to-argument mapping as recorded in the SSC (column C, Role_mapping). Each mapping has the form functor→role, e.g., PAT→Deed for *dopustit se* from the SSC class "commit / dopustit_se", see above.
   If the verb belongs to more SSC classes, create one record for each class (see, e.g., the *bouchnout-002* records with the ACT→Agent & PAT→Instrument mapping for the "bang / praštit" SSC class and the ACT→Assailant & PAT→Target mapping for the "hit / třísknout" SSC class).

4. Copy the mappings retrieved via CEV (Sect. 5.1) to column L (mapping via CzEngVallex), with aggregated PCEDT occurrences for each Argx.
   Example: for the verb *asistovat*, "assist; support" this column indicates the ACT→Arg0 mapping with 118 occurrences (102 "inherited" from the verb *support* and 16 from *assist*); further, two mapping options for PAT are identified, 169 cases of PAT→Arg1 (149 from *support* and 20 cases from *assist*) and 3 occurences of PAT→Arg2 (from *assist*), see Sect. 5.1.

5. Copy the mappings retrieved via SSC (Sect. 5.1) to column N (mapping via SynSemClass5.0), with aggregated PCEDT occurrences for each Argx.
   Example: for *asistovat* "assist; support", this column indicates ambiguous mappings of both ACT and PAT functors:

asistovat	
ACT→Protagonist	→Arg0 (166x) →Arg1 (128x) →Arg2 (1x)
PAT→Event	→Arg1 (53x) →Arg2 (295x)

Based on the retrieved mappings, the algorithm tries to resolve ambiguities:

6. If SSC and/or CEV provide an unambiguous mapping of individual PDT functors to PropBank arguments, put it to column G, Unambiguous mapping - SSC and/or CEV. This is, e.g., the case of the verb *svolat* "assemble" and its PAT functor where both CEV and SSC suggest the PAT→Arg1 mapping (with 197 occurrences collected in CEV and 418 in SSC, the later via "Event" semantic role).

7. If the mappings offered by the SSC and/or CEV lexicons are ambiguous but some prevail (based on PCEDT counts), show them in column H (Prevailing mapping - SSC and/or CEV; multiple suggestions are separated by #) and report ambiguity in column J.

[40] http://hdl.handle.net/11372/LRT-5480

Figure 2: Mapping PDT functors via the SSC roles for the class “appeal / líbit se” to PropBank arguments

appeal / **líbit se**	“Experiencer” →ACT	“Stimulus” →PAT
appeal	→PAT→Arg1 (19)	→ACT→Arg0 (12)
displease	→ACT→Arg0 (1)	→PAT→Arg1 (2)
sit	no PB mapping	no PB mapping
like	→ACT→Arg0 (57)	→PAT→Arg1 (61)
please	→PAT→Arg1 (14)	→ACT→Arg0 (3), Arg2 (4) →MEANS→Arg0 (1), Arg2 (5)
Summary for **líbit se**:	ACT→ Arg0 (58), Arg1 (33)	PAT→ Arg0 (16), Arg1 (63), Arg2 (9)

The mapping of a functor is “prevailing” whenever the number of PCEDT instances of the respective mapping is within 10 percentage points of the immediately more frequent suggestion, starting from the highest count.
Example: for *svolat* “assemble”, there are two possible mappings for ACT (both corresponding to the “Host” semantic role), namely 310 occurrences of Arg0 and just 1 occurrence of Arg1; the prevailing mapping ACT→Arg0 is suggested as the relevant mapping in column H.

8. If SSC offers an unambiguous mapping for at least some of the functors that differs from the mapping suggested by CEV, the SSC mappings go into column I (Unambiguous SSC mapping (other than CEV)) as SSC is considered more relevant due to its more “semantic” nature. If the SSC mapping is ambiguous, no suggestion is made and disagree is noted in column J.)
Example: with *střetnout_se* “compete” the mapping ACT→Arg0 unambiguously suggested in SSC with 72 occurrences in PCEDT (with the “Competitor” role) is considered as the relevant mapping and copied to column I, disregarding Arg1 mapping suggested in CEV (6 cases).

Final mappings for the whole rolesets. After the above rather bookkeeping steps (providing, at the same time, relevant background information for the annotators), the algorithm continues by deciding which suggestions to actually make to the annotators.

The suggested mapping is a union of those individual argument mappings inserted in the above steps to columns G, H and I (unambiguous, prevailing, and SSC-only mappings), fulfilling these additional “well-formed roleset” criteria:

- The indices of automatically proposed argument labels must be continuous, starting with Arg0 or Arg1 (per PropBank rules); e.g., the sequence Arg0, Arg2, and Arg3 is not a valid roleset (in such a case, the discontinuous Args note is put in column J).
Example: the valency frame corresponding to *hnát-001* “drive; force” consists of three functors, ACT for “Stimulus”, PAT for “Affected” and DIR3 for “State_final”). The mapping retrieved from the relevant SSC class “bring / dovést” suggests their correspondence to Arg0, Arg1, and Arg3, respectively, which is not considered “well-formed roleset”; thus, no final mapping is suggested. However, the annotators get highly useful information about prevailing ACT→Arg0 mapping, unambiguous PAT→Arg1 mapping, and possible mappings of DIR3 to (already taken) Arg1 (attested 19x for the given SSC class in PCEDT), Arg2 (attested 22x), and (inapt) Arg2 (attested 37x).

- The PDT-based valency frame as a whole (i.e., all its functors) must be mapped onto arguments (if not, the partial note is put in column J).
Example: with *donášet-003* “inform; snitch”, only for one functor (out of 4), possible ACT→Arg0 mapping is suggested in CEV (with 3 occurrences); no roleset is proposed.

- Argument labels do not repeat; e.g., the roleset (Arg0, Arg1, Arg1) is not a valid one (reported as repeated in column J).
Example: the valency frame of the verb *donést-002* “carry” consists of three functors, ACT for “Transporter” semantic role, PAT for “Transported” and DIR3 for “Area 2”. While in SSC, ACT is unambiguously mapped to Arg0 (40x in PCEDT) and the PAT→Arg1 mapping prevails (80x), the only suggestion for DIR3 comes from CEV, repeating Arg1. Thus no final roleset is proposed (and information on partial mappings is provided to the annotators).

Mappings that satisfy these criteria are copied to the AUTOMATIC MAPPING column (column D; the SSC-only mappings are preceded with ?). Column

K (source) contains the source of the suggested mapping (czengvallex, ssc or both).[41]

To summarize, the final output has a form of a simple table identifying, for each Czech verb sense from the PDT-Vallex lexicon (columns A, B), its functors/arguments (column B), its SSC class and semantic roles for individual functors (column C), and their automatic mapping to PropBank arguments, whenever such mapping has been considered as reliable enough (column D, with column K substantiating the decision).

Finally, columns E (CORRECTION) and F (COMMENTS) serve as the editable columns for the annotators to eventually fill in. The other columns store the source information from CEV and SSC (whenever available) plus information why it was not possible to suggest the reliable mapping automatically (where relevant, in column J).

6. Statistics And Limitations

While we cannot yet report on the amount of manual work necessary to fill in the gaps caused by the missing, ambiguous or otherwise unusable data, we present here overall statistics about the major cases, especially those mappings where the level of certainty of producing the correct mapping automatically is high.

For the individual functors, as found in the source valency lexicon, PDT-Vallex, and regardless in which valency frame they occur, the following results have been obtained:

	unamb- iguous	pref- erred	un- mapped	total
functors	9,465	8,579	24,072	42,116
percent	23	20	57	100

The above table shows that about 43 percent of arguments was possible to map to a PropBank argument label automatically with certainty (or as a preferred variant based on corpus usage statistics).

From the full roleset point of view, the situation is less favorable, albeit expected since for a valency frame to be fully mapped to a PropBank roleset, all arguments must be reliably mapped (with an avg. of 2,69 arguments per valency frame):

	auto- suggested	un- assigned	total
rolesets	5,085	10,569	15,654
percent	32	68	100

It is however important to note that most of the unassigned rolesets are simply due to missing source-side mappings (in CEV and SSC). When some mapping was available, then the problematic cases have only been a few: 117 ambiguous mappings for a functor to Argx link, 328 for noncontinuous numbering or Agrxs in the roleset, 354 for repeated Argx in a roleset, and 1,123 for only partially mapped frames.

Limitation. There is an important limitation for the approach to argument mapping as described in this paper: it needs the richly linked resources as described in the paper, in order to have reliable indications for what frames can be mapped automatically and which can only be proposed as preferred mappings, with the preferences coming from a corpus annotated by the very valency frames that have been used as a starting point.

However, the limitation might be relieved by using only one input resource, which however must at least be linked to PropBank, such as the SynSemClass one. While it can produce ambiguous or partial rolesets, and given the lack of checks against another resource, less reliable results, it can still be considered a good starting point as demonstrated by the fact that slightly more of the extracted mappings came from the SSC than from CEV (by about 200, or 1.3/3.9% from all/auto-suggested rolesets).

7. Conclusions

We have demonstrated that a carefully designed preprocessing for finding automatic mappings from a Czech valency dictionary which is based on a different theoretical approach can still produce many reliable PropBank-style rolesets (32 percent of the original full frames) to be included in a PropBank frame files for Czech. Additionally, the preprocessing produces a table (spreadsheet) with the necessary valency / predicate-argument information and clickable links for the annotators to finish the work manually in an efficient manner. In the future, the resulting Czech PropBank frame files will be used for Czech UMR annotation that follows the original guidelines requiring PropBank-style argument labels. In addition, it will also allow for more direct, large-scale comparison between the two approaches to predicate-argument labeling.

Acknowledgements

The work described herein has been supported by the grant *Language Understanding: from Syntax to Discourse* of the Czech Science Foundation (Project No. 20-16819X). It has been using data and tools provided by the *LINDAT/CLARIAH-CZ Research Infrastructure* (https://lindat.cz), supported by the Ministry of Education, Youth and Sports of the Czech Republic (Project No. LM2023062).

[41] For technical reasons, some valency frames recently edited, the older version of which should rather be deleted in the resulting roleset list (greyed rows), are marked as copy in column K.

8. Bibliographical References

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 *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*, pages 178–186.

Ondřej Bojar and Jana Šindlerová. 2010. Building a Bilingual ValLex Using Treebank Token Alignment: First Observations. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC 2010)*, pages 304–309, Valletta, Malta. ELRA.

Julia Bonn, Skatje Myers, Jens van Gysel, Lukas Denk, Meagan Vigus, Jin Zhou, Andrew Cowell, William Croft, Jan Hajič, James Martin, Alexis Palmer, Martha Palmer, James Pustejovsky, Zdeňka Urešová, Rosa Vallejos, and Nianwen Xue. 2023. Mapping AMR to UMR: Resources for adapting existing corpora for cross-lingual compatibility. In *Proceedings of the 21st International Workshop on Treebanks and Linguistic Theories*, pages 74–95, Washington, D.C., USA. Association for Computational Linguistics, Association for Computational Linguistics.

Silvie Cinková. 2006. From PropBank to EngValLex: Adapting the PropBank-Lexicon to the Valency Theory of the Functional Generative Description. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, pages 2170–2175, Genova, Italy. ELRA.

S. Cinková. 2006. From PropBank to EngValLex: adapting the PropBank-Lexicon to the valency theory of the functional generative description. In *Proceedings of LREC 2006, Genova, Italy*.

D. Dowty. 1991. Thematic Proto-Roles and Argument Selection. *Language*, 67(3):547–619.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA and London.

Cristina Fernández-Alcaina, Eva Fučíková, and Zdeňka Urešová. 2022. Annotation guidelines for Spanish verbal synonyms in the SynSemClass lexicon. Technical Report 72, UFAL MFF UK.

Eva Fučíková, Cristina Fernández-Alcaina, Jan Hajič, and Zdeňka Urešová. 2024. Textual coverage of eventive entries in lexical semantic resources. In *Proceedings of the 13th Conference on Language Resources and Evaluation (LREC-COLING 2024)*, Torino, Italy. European Language Resources Association/ICCL (to appear).

Jan Hajič, Eduard Bejček, Jaroslava Hlavacova, Marie Mikulová, Milan Straka, Jan Štěpánek, and Barbora Štěpánková. 2020. Prague dependency treebank - consolidated 1.0. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5208–5218, Marseille, France. European Language Resources Association.

Jan Hajič, Eva Hajičová, Jarmila Panevová, Petr Sgall, Ondřej Bojar, Silvie Cinková, Eva Fučíková, Marie Mikulová, Petr Pajas, Jan Popelka, Jiří Semecký, Jana Šindlerová, Jan Štěpánek, Josef Toman, Zdeňka Urešová, and Zdeněk Žabokrtský. 2012. Announcing Prague Czech-English Dependency Treebank 2.0. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 3153–3160, Istanbul, Turkey. European Language Resources Association (ELRA).

Jan Hajič, Jarmila Panevová, Zdeňka Urešová, Alevtina Bémová, Veronika Kolářová, and Petr Pajas. 2003. PDT-VALLEX: Creating a Large-coverage Valency Lexicon for Treebank Annotation. In *Proceedings of The Second Workshop on Treebanks and Linguistic Theories*, volume 9 of *Mathematical Modeling in Physics, Engineering and Cognitive Sciences*, pages 57—68, Vaxjo, Sweden. Vaxjo University Press.

Eva Hajičová and Ivona Kučerová. 2002. Argument/Valency Structure in PropBank, LCS Database and Prague Dependency Treebank: A Comparative Pilot Study. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC 2002)*, pages 846—851. ELRA.

Ishan Jindal, Alexandre Rademaker, Michał Ulewicz, Ha Linh, Huyen Nguyen, Khoi-Nguyen Tran, Huaiyu Zhu, and Yunyao Li. 2022. Universal proposition bank 2.0. In *Proceedings of the Language Resources and Evaluation Conference*, pages 1700–1711, Marseille, France. European Language Resources Association.

Paul Kingsbury and Martha Palmer. 2002. From TreeBank to PropBank. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC'02)*, Las Palmas, Canary Islands - Spain. European Language Resources Association (ELRA).

Markéta Lopatková, Václava Kettnerová, Eduard Bejček, Anna Vernerová, and Zdeněk Žabokrtský. 2016. *Valenční slovník českých sloves VALLEX*. Karolinum, Praha.

Maddalen Lopez de Lacalle, Egoitz Laparra, Itziar Aldabe, and German Rigau. 2016. A multilingual predicate matrix. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2662–2668, Portorož, Slovenia. European Language Resources Association (ELRA).

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Marie Mikulová, Alevtina Bémová, Jan Hajič, Eva Hajičová, Jiří Havelka, Veronika Kolářová, Markéta Lopatková, Petr Pajas, Jarmila Panevová, Magda Razímová, Petr Sgall, Jan Štěpánek, Zdeňka Urešová, Kateřina Veselá, Zdeněk Žabokrtský, and Lucie Kučová. 2005. Anotace na tektogramatické rovině Pražského závislostního korpusu. Anotátorská příručka. Technical Report TR-2005-28, ÚFAL MFF UK, Prague, Prague.

Roberto Navigli and Simone Paolo Ponzetto. 2012. Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Martha Palmer, Claire Bonial, and Diana McCarthy. 2014. SemLink+: FrameNet, VerbNet and event ontologies. In *Proceedings of Frame Semantics in NLP: A Workshop in Honor of Chuck Fillmore (1929-2014)*, pages 13–17, Baltimore, MD, USA. Association for Computational Linguistics.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Comput. Linguist.*, 31(1):71–106.

Jarmila Panevová. 1994. Valency frames and the meaning of the sentence. *The Prague School of Structural and Functional Linguistics*, 41:223–243.

Karin Kipper Schuler and Martha S. Palmer. 2005. *Verbnet: A Broad-Coverage, Comprehensive Verb Lexicon*. Ph.D. thesis, University of Pennsylvania, USA. AAI3179808.

Petr Sgall, Eva Hajičová, and Jarmila Panevová. 1986. *The Meaning of the Sentence in its Semantic and Pragmatic Aspects*. D. Reidel, Dordrecht.

J. Šindlerová and O. Bojar. 2009. Towards English-Czech Parallel Valency Lexicon via Treebank Examples. In *Eighth International Workshop on Treebanks and Linguistic Theories*, pages 185–195.

Jana Šindlerová, Eva Fučíková, and Zdeňka Urešová. 2015. Zero alignment of verb arguments in a parallel treebank. In *Proceedings of the Third International Conference on Dependency Linguistics (Depling 2015)*, pages 330–339, Uppsala, Sweden. Uppsala University, Uppsala University.

Kevin Stowe, Jenette Preciado, Kathryn Conger, Susan Windisch Brown, Ghazaleh Kazeminejad, James Gung, and Martha Palmer. 2021. SemLink 2.0: Chasing lexical resources. In *Proceedings of the 14th International Conference on Computational Semantics (IWCS)*, pages 222–227, Groningen, The Netherlands (online). Association for Computational Linguistics.

Zdeňka Urešová. 2011. *Valenční slovník Pražského závislostního korpusu (PDT-Vallex)*. Studies in Computational and Theoretical Linguistics. Ústav formální a aplikované lingvistiky, Praha, Czechia.

Zdeňka Urešová, Eva Fučíková, Jan Hajič, and Jana Šindlerová. 2015a. CzEngVallez – Czech–English Valency Lexicon.

Zdeňka Urešová, Eva Fučíková, Jan Hajič, and Karolina Zaczynska. 2021. Annotation guidelines for german verbal synonyms included in synsemclass lexicon. Technical Report TR-2021-70, ÚFAL MFF UK.

Zdeňka Urešová, Eva Fučíková, and Eva Hajičová. 2019a. Czengclass: Contextually-based synonymy and valency of verbs in a bilingual setting. Technical Report 62, ÚFAL MFF UK, Prague, Czechia.

Zdeňka Urešová, Eva Fučíková, Eva Hajičová, and Jan Hajič. 2018a. Creating a Verb Synonym Lexicon Based on a Parallel Corpus. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC'18)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Zdeňka Urešová, Eva Fučíková, Eva Hajičová, and Jan Hajič. 2018b. A Cross-lingual synonym classes lexicon. *Prace Filologiczne*, LXXII:405–418.

Zdeňka Urešová, Eva Fučíková, Eva Hajičová, and Jan Hajič. 2018c. Defining verbal synonyms: between syntax and semantics. In *Proceedings of the 17th International Workshop on Treebanks and Linguistic Theories (TLT 2018), Vol. 155*, Linköping Electronic Conference Proceedings, pages 75–90, Linköping, Sweden. Universitetet i Oslo, Linköping University Electronic Press.

Zdeňka Urešová, Eva Fučíková, Eva Hajičová, and Jan Hajič. 2019b. Meaning and Semantic Roles in CzEngClass Lexicon. *Jazykovedný časopis / Journal of Linguistics*, 70(2):403–411.

Zdenka Uresova, Eva Fucikova, Eva Hajicova, and Jan Hajic. 2020. SynSemClass linked lexicon: Mapping synonymy between languages. In *Proceedings of the 2020 Globalex Workshop on Linked Lexicography*, pages 10–19, Marseille, France. European Language Resources Association.

Zdeňka Urešová, Eva Fučíková, Eva Hajičová, and Jan Hajič. 2020. SynSemClass Linked Lexicon: Mapping Synonymy between Languages. In *Proceedings of the 2020 Globalex Workshop on Linked Lexicography (LREC 2020)*, pages 10–19, Marseille, France. European Language Resources Association.

Zdeňka Urešová, Eva Fučíková, and Jana Šindlerová. 2015b. Czengvallex: Mapping valency between languages. Technical Report TR-2015-58, ÚFAL MFF UK.

Zdeňka Urešová, Eva Fučíková, and Jana Šindlerová. 2016. CzEngVallex: a bilingual Czech-English valency lexicon. *The Prague Bulletin of Mathematical Linguistics*, 105:17–50.

Zdeňka Urešová, Jan Hajič, and Ondřej Bojar. 2014. Comparing czech and english AMRs. In *Proceedings of Workshop on Lexical and Grammatical Resources for Language Processing (LG-LP 2014, at Coling 2014)*, pages 55–64, Dublin, Ireland. Dublin City University, Association for Computational Linguistics and Dublin City University.

Jens E. L. Van Gysel, Meagan Vigus, Jayeol Chun, Kenneth Lai, Sarah Moeller, Jiarui Yao, Tim O'Gorman, Andrew Cowell, William Croft, Chu-Ren Huang, Jan Hajič, James H. Martin, Stephan Oepen, Martha Palmer, James Pustejovsky, Rosa Vallejos, and Nianwen Xue. 2021. Designing a Uniform Meaning Representation for Natural Language Processing. *KI - Künstliche Intelligenz*, 35(0):343–360.

Shira Wein and Julia Bonn. 2023. Comparing UMR and cross-lingual adaptations of AMR. In *Proceedings of the Fourth International Workshop on Designing Meaning Representations*, pages 23–33, Nancy, France. Association for Computational Linguistics.

Nianwen Xue, Ondřej Bojar, Jan Hajič, Martha Palmer, Zdeňka Urešová, and Xiuhong Zhang. 2014. Not an interlingua, but close: Comparison of english AMRs to chinese and czech. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, pages 1765–1772, Reykjavík, Iceland. European Language Resources Association.

Nianwen Xue, Fu-Dong Chiou, and Martha Palmer. 2002. Building a large-scale annotated chinese corpus. In *Proceedings of COLING 2002*, volume 2, pages 1100–1106, Taipei, Taiwan.

Zdeněk Žabokrtský and Markéta Lopatková. 2007. Valency information in VALLEX 2.0: Logical structure of the lexicon. *The Prague Bulletin of Mathematical Linguistics*, 2007(87):41–60.

9. Language Resource References

Hajič, Jan and Bejček, Eduard and Bémová, Alevtina and Buráňová, Eva and Fučíková, Eva and Hajičová, Eva and Havelka, Jiří and Hlaváčová, Jaroslava and Homola, Petr and Ircing, Pavel and Kárník, Jiří and Kettnerová, Václava and Klyueva, Natalia and Kolářová, Veronika and Kučová, Lucie and Lopatková, Markéta and Mareček, David and Mikulová, Marie and Mírovský, Jiří and Nedoluzhko, Anna and Novák, Michal and Pajas, Petr and Panevová, Jarmila and Peterek, Nino and Poláková, Lucie and Popel, Martin and Popelka, Jan and Romportl, Jan and Rysová, Magdaléna and Semecký, Jiří and Sgall, Petr and Spoustová, Johanka and Straka, Milan and Straňák, Pavel and Synková, Pavlína and Ševčíková, Magda and Šindlerová, Jana and Štěpánek, Jan and Štěpánková, Barbora and Toman, Josef and Urešová, Zdeňka and Vidová Hladká, Barbora and Zeman, Daniel and Zikánová, Šárka and Žabokrtský, Zdeněk. 2020. *Prague Dependency Treebank - Consolidated 1.0 (PDT-C 1.0)*. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University, `http://hdl.handle.net/11234/1-3185`.

Lopatková, Markéta and Kettnerová, Václava and Mírovský, Jiří and Vernerová, Anna and Bejček, Eduard and Žabokrtský, Zdeněk. 2022. *VALLEX 4.5*. LINDAT/CLARIAH-CZ Digital Library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University `http://hdl.handle.net/11234/1-4756`.

Mitchell P. Marcus and Beatrice Santorini and Mary Ann Marcinkiewicz and Ann Taylor. 1999. *Penn Treebank-3 (LDC99T42)*. Linguistic Data Consortium, Philadelphia, PA, USA, ISLRN 141-282-691-413-2.

Urešová, Zdeňka and Alcaina, Cristina Fernández and Bourgonje, Peter and Fučíková, Eva and Hajič, Jan and Hajičová, Eva and Rehm, Georg and Rysová, Kateřina and Zaczynska, Karolina. 2023. *SynSemClass 5.0*. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University, http://hdl.handle.net/11234/1-5230.

Zdeňka Urešová and Eva Fučíková and Jan Hajič and Jana Šindlerová. 2015. *CzEngVallex - Czech English Valency Lexicon*. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University, http://hdl.handle.net/11234/1-1512.

Urešová, Zdeňka and Štěpánek, Jan and Hajič, Jan and Panevová, Jarmila and Mikulová, Marie. 2014. *PDT-Vallex: Czech Valency lexicon linked to treebanks*. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University, http://hdl.handle.net/11858/00-097C-0000-0023-4338-F.

Lexicalized Meaning Representation (LMR)

Jorge Baptista[1,2], **Sónia Reis**[1,2], **João Dias**[1,2,3], **Pedro A. Santos**[2,4]
[1]U. Algarve - FCHS/FCT, [2]INESC-ID LIsboa, [3]CISCA, [4]U. Lisboa - IST
Faro/Lisboa, Portugal
jbaptis,smreis,jmdias@ualg.pt, pedro.santos@tecnico.ulisboa.pt

Abstract

This paper presents an adaptation of the Abstract Meaning Representation (AMR) framework for European Portuguese. This adaptation, referred to as Lexicalized Meaning Representation (LMR), was deemed necessary to address specific challenges posed by the grammar of the language, as well as various linguistic issues raised by the current version of AMR annotation guidelines. Some of these aspects stemmed from the use of a notation similar to AMR to represent real texts from the legal domain, enabling its use in Natural Language Processing (NLP) applications. In this context, several aspects of AMR were significantly simplified (e.g., the representation of multi-word expressions, named entities, and temporal expressions), while others were introduced, with efforts made to maintain the representation scheme as compatible as possible with standard AMR notation.

Keywords: Lexicalized Meaning Representation (LMR), Abstract Meaning Representation (AMR), Natural Language Processing (NLP), Portuguese

1. Introduction

This paper aims to contribute to the development of a theoretical and formal framework for the semantic annotation of natural language texts, facilitating the creation of tools for computational language processing. Semantic annotation of natural language texts aims to establish a representation of meaning that is valuable for developing various tools and applications (Damonte et al., 2017; Damonte and Cohen, 2018; Seno et al., 2022), particularly in Natural Language Processing (NLP). These applications include automatic sense disambiguation, machine translation, text summarization, and the generation of multilingual documents.

Various initiatives have been developed for this purpose. The Universal Networking Language (UNL) (Uchida et al., 1996)[1] provided a version of the novella *The Little Prince* (TLP) by Antoine de Saint-Exupéry (Martins, 2012) with the explicit aim of comparing representations of the same text in different languages. More recently, Abstract Meaning Representation (AMR) (Banarescu et al., 2013) has gained popularity in the NLP community. Originally proposed for English, this model aims to represent the meaning of sentences in a simplified form.

In a nutshell, each sentence's meaning is represented as a directed acyclic graph without a root. In this graph, nodes correspond to semantic predicates (operators) and their arguments, while arcs represent the semantic relations between the sentence elements. These relations, known as semantic roles, are defined in *OntoNotes* (Weischedel et al., 2013) and are associated with the arguments of (mostly) verbal predicates.

The frames of these verbal predicates form an ontology acting as a 'catalog' of meanings, serving as a reference for the various meanings of predicative elements represented in the graph. Additionally, other semantic relations are expressed by labeled arcs, linking predicates to different types of elements and circumstances, sometimes replacing textual elements that convey these relations. Grammatical elements such as auxiliary verbs, copulas, or support verbs are simply omitted. Many lexical elements are replaced either by verbs listed in *OntoNotes* or by other elements (e.g., adverbs ending in *-ly* are replaced by the morphologically associated adjectives and linked to an operator by the labeled arc :MANNER). Figure 1 illustrates the standard AMR graph representation of a simple English sentence extracted from the mentioned novella – *Draw me a sheep ...* [TLP:id=65], produced by the AMREager parser (Damonte et al., 2017)[2].

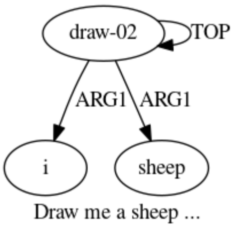


Figure 1: Standard AMR graph

The graph representing the meaning of the sentence can also be built in an equivalent PENMAN formalism (Matthiessen and Bateman, 1991). Such a PENMAN graph is shown in Figure 2 taken from the AMR annotation of the novella.[3] (The differ-

[1] http://www.unlweb.net/

[2] https://bollin.inf.ed.ac.uk/amreager.html

[3] https://amr.isi.edu/download/

```
(d / draw-01
    :ARG0 (y / you)
    :ARG1 (s / sheep)
    :ARG2 (i / i)
    :mode imperative)
```

Figure 2: AMR graph in PENMAN formalism

ences between the PENMAN graph and the AMR parser's output are deemed irrelevant for the purpose of this paper.)

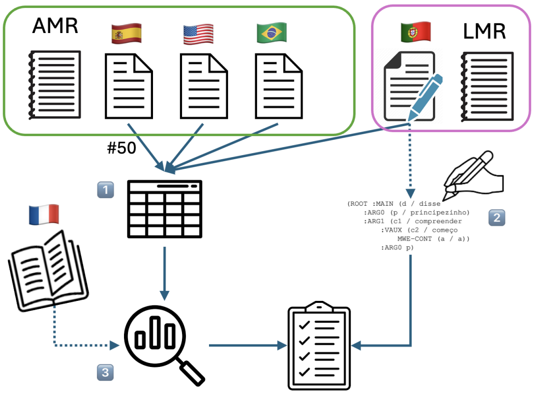
Although AMR has been initially conceived for the English language and explicitly rejects the classification of inter-language (Banarescu et al., 2013), it naturally lends itself to the comparison of annotations of the same text in various languages (Xue et al., 2014). This annotation scheme was, rightly, adapted for the representation of texts, either by different annotators, or translations of the same text in different languages. Examples include annotations of the same novella in English, Chinese (Li et al., 2016), Spanish (Migueles Abraira, 2017), Turkish (Azin and Eryiğit, 2019; Oral et al., 2024), Vietnamese (Linh and Nguyen, 2019), Brazilian Portuguese (Anchiêta, 2020), and Persian (Takhshid et al., 2022).

On the other hand, the initial version of AMR aimed at describing individual sentences independently. Recently, however, the AMR guidelines were extended to the Unified Meaning Representation (UMR) formalism (Pustejovsky et al., 2019; Wein and Bonn, 2023) to encompass the annotation of sequences of sentences forming discourses (O'Gorman et al., 2018). Naturally, the original guidelines were occasionally reviewed and expanded to incorporate concepts that had not been sufficiently considered in the original proposal (Bonial et al., 2018). As recently mentioned by (Seno et al., 2022), following (Hovy and Lavid, 2010), these reformulations and extensions seek to achieve "the necessary balance between the depth of linguistic theory to be used and the stability of the annotation process", which does not prevent "critics within the community interested in this semantic representation, regarding some decisions made originally" (Seno et al., 2022, p. 51).

The primary objective of this work is to establish an annotation scheme, inspired by the AMR guidelines[4], which aims to address a set of difficulties and problems encountered in the solutions adopted thus far (see Section 2 and Table 1 for an overview).

To achieve this goal, we compared available Abstract Meaning Representation (AMR) annotations from parts of Antoine de Saint-Exupéry's work *The Little Prince* in English, Spanish (Migueles Abraira, 2017)[5], and Brazilian Portuguese (Anchiêta, 2020)[6], along with a Lexicalized Meaning Representation (LMR) annotated version of the same work in European Portuguese. We occasionally consulted the original French edition of the novella to verify any changes introduced by the translators.

Our focus was on the 50 Spanish sentences translated from the English version by Migueles Abraira (2017), ensuring a 4-tuple comparison. We conducted a critical analysis of these 50 sentences, considering observed phenomena and the annotation solutions adopted, comparing the similarities and differences between the annotations. Note that the translators' choices regarding the Portuguese or Spanish sentences are not considered here. Instead, the focus is solely on the structure and meaning of the translation output and the corresponding semantic representation (AMR/LMR). Due to space constraints, this paper provides only a succinct overview highlighting the main findings.



```
(ROOT :MAIN (d / disse
    :ARG0 (p / principezinho)
    :ARG1 (c1 / compreender
        :VAUX (c2 / começo
            MWE-CONT (a / a))
        :ARG0 p)
```

Figure 3: Comparing AMR/LMR annotations: alignment, annotation and analysis.

Figure 3 outlines the procedural stages of this study: (1) Alignment of sentences in different languages/varieties, considering translations from the original edition of the work (the English edition in the case of the Spanish version; the French edition in the case of the Portuguese translations) and resolving encountered alignment mismatches. (2) Annotation of sentences in the European Portuguese version of *The Little Prince* (Baptista, 2024b)[7], inde-

amr-bank-struct-v3.0.txt

[4] AMR 1.2.6. Specification (2019): https://github.com/amrisi/amr-guidelines/blob/master/amr.md

[5] https://github.com/ixa-ehu/amr-corpus-spanish/blob/master/es-Little-Prince-Corpus-50-AMR.txt

[6] https://github.com/rafaelanchieta/amr-br/blob/master/amr_br-v1.0.xml

[7] The European Portuguese, LMR-annotated sentences can be found at: https://gitlab.hlt.inesc-id.pt/u000803/lmr4pt/-/blob/master/public/LMR4PT_Principezinho.pdf.

pendently performed by two annotators and based on a set of LMR Guidelines autonomously developed by Baptista (2024a). These guidelines aim to: (a) adapt AMR to linguistic situations observed in Portuguese but not observed in English; (b) systematically make explicit and consistent the relation between text elements and annotation; and (c) adequately account for relevant linguistic phenomena not contemplated by the AMR framework. (3) Finally, a critical and systematic comparison of the annotations of sentences in the different languages was conducted.

2. Comparing AMR and LMR

Considering specific aspects of European Portuguese, as well as other fundamental requirements of semantic annotation, LMR introduces several extensions and reformulations of the standard AMR annotation scheme proposed by (Banarescu et al., 2013). Table 1 schematically presents the main differences between AMR and LMR annotations.

The Abstract Meaning Representation (AMR) annotation scheme is grounded in a 'catalog' of meanings derived from the verbal constructions present in *OntoNotes* (Weischedel et al., 2013). This methodology accepts both the reconstruction and suppression of textual elements, such as the insertion of pronouns in lexically unfilled syntactic positions or the replacement of conjunctions and prepositions with the semantic relations they convey. However, it does not encompass the analysis of auxiliary verbs, including copulative verbs (*Vcop*) and support verbs (*Vsup*) (or so-called *light* verbs). In fact, only some constructions with *Vsup* are considered, as most predicative nouns are assimilated into the corresponding verbal predicates (for example, [*a*] *purchase* → [*to*] *buy*). Additionally, AMR represents complex named entities (NE), particularly for denoting temporal and quantity values.

Lexicalized Meaning Representation (LMR), on the other hand, emphasizes a representation closely tied to the text, effectively constituting an annotation process where representation is directly anchored on the words of the sentences rather than merely appended to them as a whole. Furthermore, LMR strictly adheres to the principle of not replacing words in the text with arbitrary or theoretical constructs, instead anchoring relations on surface forms — the only visible elements that provide access to the meaning of the sentence[8].

As a semantic ontology or 'catalog' of word senses, LMR relies on the *Dicionário Gramatical de Verbos do Português* [Grammatical Dictionary of Portuguese Verbs] (DGVP; Baptista and Mamede, 2020a), built on the database of the lexicon-grammar of European Portuguese verbs (ViPEr; Baptista, 2012; Baptista, 2013 (Baptista and Mamede, 2020c)), as the 'catalog' of meanings of verbal constructions. For nominal predicates, the lexicon-grammar of predicative nouns (SNIPER; Baptista and Mamede, 2020b) is used. Since both resources indicate adjectival counterparts of these verbal and nominal predicates, and in the absence of a lexicon-grammar of adjectives proper for Portuguese, the adjective is referenced to either one or the other (or both) (for simplicity, these references were not provided in this paper).

For a semantic ontology or 'catalog' of word senses, LMR relies on the *Dicionário Gramatical de Verbos do Português* (DGVP; Baptista and Mamede, 2020a), which is built on the database of the lexicon-grammar of European Portuguese verbs (ViPEr; Baptista, 2012; Baptista, 2013), serving as the 'catalog' of meanings of verbal constructions. For nominal predicates, the lexicon-grammar of predicative nouns (SNIPER; Baptista and Mamede, 2020b) is utilized. Since both resources indicate adjectival counterparts of these verbal and nominal predicates, and in the absence of a lexicon-grammar of adjectives specific to Portuguese, the adjective is referenced to either one or the other (or both) (for simplicity, these references were not provided in this paper).

One of the major differences, thus, between AMR and LMR is that LMR adopts a homologous strategy for representing the predicate-argument relations from different grammatical categories, that is, verbs, nouns and adjectives. In this way, words in the texts are represented in LMR respecting their part-of-speech, keeping the representation closer to the text. For example, the sentence TLP id=348 is represented in AMR as shown in Figure 4:

One of the major differences, therefore, between AMR and LMR is that LMR adopts a homologous strategy for representing the predicate-argument relations from different grammatical categories — verbs, nouns, and adjectives. This approach ensures that words in the texts are represented in LMR according to their part-of-speech, maintaining a representation closer to the text. For example, the sentence TLP id=348 is represented in AMR as shown in Figure 4.

In this case, the adjective *important* in under the main predicate *think*, and the copula verb *be* is ignored. In turn, in the corresponding Portuguese sentence: *– Isso não é importante?!* 'That is not important?!' [TLP id=348], the copula verb is linked to the adjective it auxiliates (Figure 5):

[8] Certain types of zeroing, such as *appropriate* zeroing (Harris, 1976, 1991), pose serious challenges to this approach, for example, *John enjoyed* (*reading*) *the book*. These challenges must be addressed differently, though they are outside the scope of this paper.

Abstract Meaning Representation (AMR) (Banarescu *et al*. 2013)	**Lexicalized Meaning Representation (LMR)**
A catalog of senses (semantic predicates) for verbs can be found in OntoNotes by Weischedel, R. et al. (2013). Other categories such as nouns and adjectives are represented by verbal predicates.	A catalog of senses is available in the Lexicon-Grammar of Portuguese. For verbs, references include ViPEr by Baptista (2012, 2013) and the Dictionary of Portuguese Verb Grammar by Baptista & Mamede (2020a). Predicative nouns are covered in SNIPER by Baptista & Mamede (2020b).
Directed acyclic graphs lack a root node, instead employing an arc labeled :TOP looping over the main predicative element node of the sentence.	Directed acyclic graphs feature a ROOT node, which is connected to the main predicative element (:MAIN), serving as the node to which elements with scope over the entire sentence are connected.
Reduced elements are reconstructed.	No reconstruction of reduced elements is performed.
A graph representation is appended to the entire sentence, without establishing a direct relation between the graph nodes and the text forms.	There exists an explicit relation between text forms and their representation, treating text forms as nodes of the graph.
Predicative elements in the text are replaced by verbal lemmas (especially verbs represented in OntoNotes).	Predicative elements in the text are preserved in the graph, with the association of lemmas and constructions being carried out in the post-processing phase.
Some textual elements undergo substitution, especially grammatical ones (such as conjunctions, prepositions, etc.), by the semantic relations they express.	All textual elements undergo maintenance, alongside explicit representation of the semantic relations they convey; these include conjunctions, prepositions, subordinate gerund *-ndo* '-ing' morpheme, etc.
Auxiliary verbs, copulative verbs, or support verbs (light verbs) are not considered.	All types of auxiliary verbs are considered, including verbal auxiliaries (temporal, modal, and aspectual), adjectival auxiliaries (copulative verbs), nominal auxiliaries (support verbs), and auxiliaries of passive constructions. Additionally, constructions with (causative, linking and agentive) operator verbs are also taken into account.
Multi-word expressions (MWE) of varying complexity are represented, with a sophisticated representation of named entities (NE), and particularly temporal and quantification expressions.	Very simplified representation of multi-word expressions (MWE), named entities (NE), as well as temporal and quantification expressions. MWE and NE are identified in the pre-processing phase and integrated as nodes in the LMR graph.
Intra-phrasal anaphoric relations are represented, alongside an extension of notation (O'Gorman et al., 2018) for trans-phrasal anaphoric relations through coreference chains at the text level.	Intra-phrasal anaphoric relations are represented solely between explicit elements in the text, with anaphora resolution addressed as a post-processing task (trans-phrasal anaphoric relations are not yet considered).
Verbal predicates (standard representation) and adjectival (:DOMAIN) are treated distinctly, while nominal constructions are represented by verbal constructions if present in OntoNotes.	Verbal, nominal, and adjectival predicates feature a homologous representation of argument structure, corresponding to the standard representation: *predicate* (:ARG0, :ARG1, ...).

Table 1: Summarized comparison between Abstract Meaning Representation (AMR) and Lexicalized Meaning Representation (LMR)

You think that is not important ! . [id=348]

```
(t / think-01
   :ARG0 (y / you)
   :ARG1 (t2 / that
      :ARG1-of (i / important-01
         :polarity -)))
```

Figure 4: AMR Representation of sentence id=348

Isso não é importante ?! 'That is not important?!' [id=348]

```
ROOT :MAIN (i1 / importante
     :VAUX (ser / é)
     :NEG (n / não)
     :ARG0 (i2 / isso))
     :MODE-EXCLAMATIVE)
```

Figure 5: AMR Representation of sentence id=348

When the main predicative element is the corresponding predicative noun *importância* 'importance', it appears in an equivalent support verb construction with support verb *ter* 'have', represented by LMR as shown in Figure 6.

Notice that the role of the negation adverb is explicitly encoded and attached to the negation adverb *não* 'not'. This solution, however, is arguably equivalent to the AMR notation, though it avoids zeroing the negation adverb and anchors the negation con-

Isso não tem importância . lit.: 'That doesn't have importance'

```
ROOT :MAIN (i1 / importância
    :VSUP (t / tem)
    :NEG (n / não)
    :ARG0 (i2 / isso)))
```

Figure 6: LMR Representation: a predicative noun in a support-verb construction

struct on a textual element. Notice also that the exclamative mode of the sentence is attached to the `:ROOT` node, which is theoretically seen here as a more adequate representation (Harris, 1991) as it bears on the entire sentence. The lack of a root node in AMR forces the modality to be attached to the main predicative element (though the AMR notation, shown in Figure 4, fails to do).

A similar representation is also proposed for the corresponding verb, if it exists in the language (these triplets are not rare in Portuguese), e.g. – *Isso não importa?!* 'That [does] not matter?!':

`(i1 / importa ... :ARG0 (i2 / isso)).`

LMR maintains the equivalence relation between lexical elements by offering analogous representations for full verbs, predicative adjectives, and predicative nouns. It maintains notation closely tied to the text, anchoring semantic representation directly on its elements. While these paraphrastic equivalence relations (*transformational*, in the sense of Harris (1964, 1976, 1991)) should indeed be established, they are better suited for higher-order representation to minimize *ad hoc* interpretations during human annotation. Ideally, the "catalog of senses" or semantic predicates underlying the AMR/LMR notation should provide such equivalence. This is indeed the case for the works by Baptista and Mamede (2020a,c).

A notable contrast between the two schemes is that in LMR, a root node (`ROOT`) is instantiated for each sentence, with a `:MAIN` dependency linking this node to the main predicative element. This resolves a technical issue previously highlighted by Anchiêta (2020) regarding the evaluation of competing semantic representations for the same sentence. Still, this also affects the adequacy of representing elements that operate on the entire sentence, such as sentence-external adverbial modifiers, as defined by Molinier and Levrier (2000). For instance, in the sentence *But my drawing is certainly very much less charming than its model* [TLP id=52], the adverb *certainly* imparts a modality value to the entire sentence, akin to *It is certain that my drawing is very much less charming than its model*. In such cases, and unlike AMR that hinges the `:mod (c / certain)` under another node of the graph, LMR suggests representing the adverb as a modifier on the ROOT node:

`(ROOT :MOD (c / certainly) ...`

By closely adhering to the text and preserving the words' part of speech, LMR effectively distinguishes between the main types of adverbial constructions: sentence-external and sentence-internal adverbs. Moreover, astute readers may have observed the conjunction *but* at the sentence's outset, serving to connect it with preceding discourse in a manner akin to *conjunctive adverbs* (or *discourse connectives*). Consequently, the identical descriptive approach is employed for both scenarios.

`(ROOT :MOD (b / but) ...`

`(ROOT :MOD (b / furthermore) ...`

Furthermore, LMR incorporates auxiliary verbs, encompassing copulative and support verbs, into its analysis. This inclusion is justified by the significance attributed to these elements as integral components of textual meaning units. Indeed, Portuguese features a particularly rich system of auxiliary verbs (Baptista et al., 2010; Baptista and Crismán Pérez, 2021), particularly for expressing aspectual nuances. For instance, in the sentence: – *Começo a compreender, disse o principezinho.* 'I begin to understand, said the little prince.' [TLP id=1080], the auxiliary *começar a* 'begin to' is represented as:

```
(ROOT :MAIN (d / disse
    :ARG0 (p / principezinho)
    :ARG1 (c1 / compreender
        :VAUX (c2 / começo
            MWE-CONT (a / a))
        :ARG0 p)
```

The auxiliary construction is depicted as a multiword expression, with a `:MWE-CONT` arc linking the auxiliary verb to the preposition it introduces. This enables distinguishing its precise aspectual value from other nuanced constructions involving the same verb but with a different preposition (e.g., *começar por* for 'begin by'). The representation of modal auxiliaries is particularly relevant for legal domain texts, where deontic modality is essential. Two domain-specific relations were devised solely for this purpose, `:dever` 'must/ought' and `:poder` 'may/can', corresponding to the verbs most commonly used with that function. Besides, a similar notation was devised for all types of auxiliary verbs. In many situations, it is possible to keep LMR compatible with AMR (except for modal auxiliaries, treated as full predicates in AMR).

LMR also adopts a simplified representation both of multi-word expressions (e.g., compound nous, idioms) and of named entities (e.g. people, organizations, and places), as well as temporal and quantification expressions, delegating this task to a pre-annotation step, prior to the semantic annotation.

Other differences of detail were envisaged. For instance, in relative sub-clauses, e.g. *the girl who* *adjusted the machine*, while AMR eliminates the relative pronoun:

```
(g / girl
     :ARG0-of (a / adjust-01
          :ARG1 (m / machine)))
```

LMR keeps the relative pronoun in the representation, maintaining consistency in the representation of the predicate-argument structure of sub-clause's predicate:

```
(g / girl
     :ARG0-of (a / adjust-01
          :ARG0 (w / who)))
          :ARG1 (m / machine)))
```

An aspect of language-specific adaptation is the existence of the so-called gerundive reduced subclauses. Here, we analyse the gerund morpheme (the *-ndo* '-ing' verb ending) as having a function similar to that of an adverbial subordinative conjunction, but with an underspecified semantic value. In fact, the nexus between the main clause and the gerundive subclause is often difficult to determine (cause, time). In order not to 'force' any interpretation, a generic `:NDO` is proposed (Figure 7).

O vaidoso recomeçou a agradecer, tirando o chapéu.
'The vain person started to thank again, tipping his hat.' [TLP id=620]

```
(ROOT :MAIN (a / agradecer
    :VAUX (r / recomeçou
        :MWE-CONT (a / a))
    :ARG0 (v / vaidoso)
    :NDO (t / tirando
        :ARG0 v
        :ARG1 (c / chapéu))))
```

Figure 7: Gerundive subclauses and :NDO

In the Brazilian Portuguese annotation of the same construction, one finds either the `:subevent-of` relation[9], or `:manner`, or even an `:arg2-of` (id=344). AMR deals with English similar gerundive sub-clauses (for example, id=631) in the same way as with relative subclauses, v.g. *"I admire you," said the little prince, shrugging his shoulders slightly,* ...:

```
(s / say-01
   :ARG0 (p / prince :mod (l / little)
      :ARG0-of (s2 / shrug-01
      :ARG1 (s3 / shoulder :part-of p)
      :degree (s4 / slight))) ...
```

This is not, arguably, a representation exactly equivalent to the meaning that the gerund subordinate operator *-ing* introduces in the sentence (two simultaneous actions). In fact, the equivalent relative clause would be: *The prince* that shrugged his shoulders *said "I admire you"* ...

On the other hand, the gerund bound morpheme is, in fact, present in the sentence, and in spite of not being able to "detach" it from the base (or host) verb, its value, vague as it is, is made explicit with the notation `:NDO`.

These methodological differences between AMR and LMR result from partly distinct approaches in the semantic representation of texts: although each presents its specific advantages and challenges, LMR distinguishes itself by seeking to reconcile the precision of semantic representation and fidelity to the underlying text, suggesting a potentially more precise approach in semantic analysis.

3. Contrastive analysis

To illustrate the systematic contrastive analysis of the notations of *The Little Prince* in the four languages here considered, we present a case study by commenting on the following sentence with id=300:

FR: *J'étais très soucieux car ma panne commençait de m'apparaître comme très grave, et l'eau à boire qui s'épuisait me faisait craindre le pire.*

In the English version, this sentence is split into two (id=299 and id=300), which we present below.

EN: *I was very much worried, for it was becoming clear to me that the breakdown of my plane was extremely serious. And I had so little drinking-water left that I had to fear for the worst.*

In the case of the Spanish translation (Migueles Abraira, 2017), which faithfully follows the English version, only the AMR representation of the second sentence is available.

ES: *Y me quedaba tan poca agua potable que me temía lo peor.* [SP id=15]

For Brazilian Portuguese (Anchiêta, 2020), which was based on the French version of the text, we find a very losely translated equivalent sentence:

BR: *Minha pane começava parecer demasiado grave, e em, breve já não teria água para beber ...*

Finally, for European Portuguese, the translator faithfully follows the French original:

PT: *Estava bastante inquieto, pois a avaria começava a parecer grave, e a pouca água que restava para beber fazia-me temer o pior.*

We start the analysis by commenting the standard AMR representation, made for the English version (Figure 8).

The first observation is the replacement of the causal subordinated conjunction *for* by the abstract construct `cause-01`. This construct takes the following arguments: as `:ARG0`, the causal subordinate clause (*it was becoming clear to me that* ...); and as `:ARG1` the main clause (*I was very much worried*).

[9] https://www.isi.edu/~ulf/amr/lib/amr-dict.html#:subevent

I was very much worried, for it was becoming clear to me that the breakdown of my plane was extremely serious. And I had so little drinking-water left that I had to fear for the worst. [EN id=299.300]

```
(c2 / cause-01
  :ARG0 (c / clear-06
    :ARG1 (s / serious-02
      :ARG1 (b / break-down-12
        :ARG1 (p / plane
          :poss i))
        :degree (e / extreme))
    :ARG2 (i / i))
  :ARG1 (w / worry-01
    :ARG1 i
    :quant (m / much
    :degree (v / very))))

(a / and
  :op1 (h3 / have-degree-91
    :ARG1 (w / water
      :purpose (d / drink-01)
  :ARG1-of (l2 / leave-17)
  :ARG1-of (h / have-03
    :ARG0 (i / i)))
  :ARG2 (l / little)
  :ARG3 (s / so)
  :ARG6 (o / obligate-01
    :ARG1 i
    :ARG2 (f / fear-01
      :ARG0 i
      :ARG1 (t / thing
        :ARG1-of (h2 / have-degree-91
          :ARG2 (b / bad-07)
          :ARG3 (m / most)))))))
```

Figure 8: English AMR representation of sentence id=299.300

In the case of the adjectival construction of `clear-06`, where a subject clause is extraposed, the subject is linked by an `:ARG1` arc, as indicated in the directives[10]. However, the 3-argument frame of `clear-06` (a verb?) had been defined with a "cause" role for its `:ARG0` (?), which is now expressed by an independent node `cause-01`.

On the other hand, representing the construction of `serious-02` as a predicate with only one argument – *something is serious* – raises difficulties in justifying the semantic relation of `:ARG1` to the subject (`break-down-12`) of this adjective. In the Ontonotes[11], `serious-02` does not even have an `ARG0` role. This highlights how the association of adjectival predicates with verbal lemmas may not be entirely appropriate. The notation of these arguments as `:ARG1` is more of an artifact of the Ontonotes representation scheme than a regular (and generalizable) configuration between semantic predicates and their arguments.

In the case of the adjectival predicate `worry-01` (*worried*), such perplexity does not arise. Its predicative structure could effectively be described by the corresponding verbal construction, given its classification as a so-called 'psychological' verb (class 04, (Baptista and Mamede, 2020a)). This would correspond to the structure *something cause somebody to worry* = *something worries somebody*. In this construction, the verb exhibits a *causative* subject and an *experiencer* complement, filled by a human noun, here represented by the pronoun *I*, to which the `:ARG1` relation could correspond.

Regarding the second sentence, the AMR annotation relies on an abstract conceptualization of predicates such as `have-degree-91`[12], which is associated with adjectival constructions expressing gradable predicates, and `have-03`[13], corresponding to the full verb *have* in the sense of "possession". However, interpreting the representation of this sentence, simplified below, remains challenging:

```
h3 / have-degree-91
  :ARG1 (w / water
    :ARG1-OF (h / have-03
      :ARG0 (i /i)))
```

This configuration does not match the sentence we are analyzing: we encounter the verb *have* with the object *water*, quantified by *so little*. Moreover, the second verb *have* (`have-03`) typically represents the meaning associated with 'possession', making the presence of both operators appear redundant, at the very least.

In the sentence *I had to fear*, the modal auxiliary *have* is replaced by the operator `obligate-01`. However, this replacement ignores the nature of the modal auxiliary, which, being transparent to the selection restrictions of the main verb *fear*, should have the same subject as this verb. Consequently, the operator appears with its subject marked as an `:ARG1`, a consequence of the substitution of the auxiliary by `obligate-01`.

Lastly, the expression *fear for the worst* is represented in a manner that attempts to analyze its idiomatic value, rather than recognizing its non-compositional semantics, which is already lexicalized.

Regarding the sentence in Spanish, corresponding only to the second sentence of the English version (id=300), the notation closely follows the standard AMR representation, as usual (Figure 9).

The conjunction *y* (and) is used here to connect the current sentence to the previous one. However,

[10] https://amr.isi.edu/doc/amr-dict.html#:domain

[11] https://propbank.github.io/v3.4.0/frames/serious.html#serious.02

[12] https://amr.isi.edu/doc/amr-dict.html#have-degree-91

[13] https://propbank.github.io/v3.4.0/frames/have.html#have.03

Y me quedaba tan poca agua potable que me temía lo peor. [SP id=15]

```
(y2 / y
  :op1 (c / causar
    :ARG0 (q / quedar
      :ARG1 (a / agua
        :mod (p / potable)
        :mod (p2 / poco
          :grado (t / tan)))
      :ARG2 (y / yo))
    :ARG1 (t / temer
      :ARG0 y
      :ARG1 (m / malo
        :grado (m2 / máximo)))))
```

Figure 9: Spanish AMR representation of sentence SP id=15

this conjunction is treated like any other coordination situation. Since there is no second coordinated element, only one conjunctive operator `:OP1` is given The operator `:OP1` should connect the conjunction to the first member of the coordination. No second member of the coordination exists, since it is the entire sentence that is being put in relation to a previous discourse. Now, accepting this to be the function of *y* (as well as that of *and*, in the English version), the first member of the coordination should be the previous sentence. As AMR does not currently handle this type of cross-sentential relations (but see (O'Gorman et al., 2018)), any notation would always be incomplete. Nevertheless, the choice of `:OP1` seems somewhat ambiguous.

Another interesting aspect is the simplification (and closer adherence to the text) of the representation of the constituent *tan poca agua potable* 'so little drinking water', an argument of *quedar* 'to be left', which is based on the words of the text and does not resort to the type of constructs seen in standard AMR. Nevertheless, we analyze this *quedar* construction as a predicate with two arguments, where *agua* 'water' should correspond to the `:ARG0`, while the first-person dative pronoun *me* corresponds to an `:ARG1`.

Finally, as in English, the annotator intended to represent the expression *lo peor* 'the worst', making it corresponds to elements that are not present in the text (*malo máximo*).

Now, let's examine the analysis of the translation in Brazilian Portuguese, comparing it with the original French version. In this sentence, the translator omitted the main clause, with the predicate *soucieux* 'worried' and the causal conjunction *car* 'for' that links it to the rest of the sentence. Similarly, there was a profound transformation of the second subordinate clause under *car*: *et l'eau à boire qui s'épuisait me faisait craindre le pire* is translated as *e em, breve já não teria água para beber...* The construction with the operator-verb *faire* 'to make' disappears, as well as the construction of the verb *s'épuiser* 'to run out/exhaust'. The idiomatic expression *craindre le pire* 'to fear the worst' also disappears. In this case, this creative translation does not allow for a direct comparison between the annotation solutions adopted among the different languages, but only a generic comment on the AMR representation produced (Figure 10).

Minha pane começava parecer demasiado grave, e em, breve já não teria água para beber ... [BR id=299;300]

```
(c / começar-01
  :ARG0 (p / pane
    :poss (m / minha))
    :ARG1 (p1 / parecer-01
      :ARG2 (g / grave
        :degree (d / demasiado)))
    :cause (t / ter-01 :polarity -
      :ARG0 (e / eu)
      :ARG1 (a / água)))
```

Figure 10: Brazilian Portuguese AMR representation of sentence id=300

Let's start by noting the treatment of *começar* 'begin', here an auxiliary verb of *parecer* 'seem', as well as the verb *parecer* itself, that are represented as full verbs. It is difficult to entertain the idea of *começar* and *parecer* as full verbs, deviating from the more conventional analysis as a copulative verbs in an adjectival construction. The relation (`:ARG2`) between this verb *parecer* and the adjective *grave* 'serious' presents an even greater challenge to comprehension.

As previously mentioned, the principle of distributional transparency of auxiliaries regarding the selection restrictions imposed by the elements they 'modify' (Baptista et al., 2010; Baptista and Crismán Pérez, 2021) suggests an analysis in which *grave* 'serious' functions as the main predicative element of this clause, with *pane* 'breakdown' as its :DOMAIN, as follows, while consistency with AMR guidelines would lead to eliminate both copula verbs:

```
(g / grave :DOMAIN (p / pane))
```

The second interesting aspect is that the coordinative conjunction *e* 'and' has been removed and replaced bay a causal relation, as denoted by the operator `:CAUSE`. While not implausible, this interpretation seems unmotivated. Finally, note the suppression of the temporal adverbial phrase *em breve* 'soon', without any apparent reason.

Finally, let's look at the translation in European Portuguese and the proposal for its annotation in LMR (Figure 11). This translation is much more 'faithful' to the original French version, only taking the liberty to modify *l'eau à boire qui s'épuisait* into

Estava bastante inquieto, pois a avaria começava a parecer grave, e a pouca água que restava para beber fazia-me temer o pior. [PT id=300]

```
ROOT :MAIN (i / inquieto
  :ARG0 m
  :VAUX (e / estava)
  :DEGREE (b1 / bastante)
  :CAUSE (p1 / pois
    :OP2 (e / e
      :COORD1 (g / grave
        :ARG0 (a / avaria)
        :VAUX (p2 / parecer
          :VAUX (c / começava
            MWE_CONT (a / a))))
      :COORD2 (f / fazia
        :CAUSE (a / água
          :QUANT (p3 / pouca)
          :ARG0-OF (r / restava
            :ARG0 (q / que)
            :PURPOSE (p4 / para
              :OP2 (b2 / beber)))
        :VOPC (top / temer_o_pior
          :ARG0 (m / me))))))
```

Figure 11: European Portuguese LMR representation of sentence id=300

a pouca água que restava para beber. This modification alters the dependency of the verb *beber* 'to drink' and inserts the quantifier *pouca* 'little' associated with the use of the verb *restar* 'to be left'. This sentence allows us to present several interesting aspects of the LMR annotation scheme. Firstly, the use of the `:OP2` operator, 'repurposed' from standard AMR to link the conjunctions *pois* 'for' and *para* 'to' to the sentences they introduce. Since the precise semantic value these conjunctions convey are (mostly) lexically determined, LMR keeps the conjunctions and the link they establish between the main clause and the sub-clause. Notice that standard AMR notation simply abstract away from the conjunction proper.

A second aspect is the explicit representation of coordination relations using the `:COORD1` and `:COORD2` operators, rather than the generic `:OP1` and `:OP2` in standard AMR. These `:COORD` operators fulfill the same function, maintaining close parallelism between the two notations.

We also analyze the verb *parecer* 'seem' following a fairly traditional approach, as a copulative verb, i.e. an auxiliary of the adjective *grave* 'serious' and the recursive auxiliary verb chain *começar a parecer* 'begin to seem'.

Another notable aspect is the treatment of relative clauses. These are connected by linking the antecedent of the relative pronoun to the verb of the relative clause via an 'inverted' `ARGn-OF` relation, where 'n' denotes the semantic relationship of this element in the base clause of the relative. Subsequently, this relation is reiterated, without inversion, between the verb of the relative clause and the relative pronoun.

Lastly, we introduce the concept of the *causative operator-verb* (*Vopc*; (Gross, 1981), (Baptista, 2005, 202 ff.)). This concept entails an operator applied to a sentence, augmenting it with an additional argument, and establishing a causal relation between this extra argument and the base sentence. In our example, the verb *fazer* (to make) fulfills this function: *A água fazia*/Vopc # *eu temia o pior* (The water made/I feared the worst). For such operators, LMR suggests delineating two relations: firstly, `:CAUSE`, connecting the operator-verb to its subject; secondly, the relation `:VOPC`, linking the operator-verb to the embedded sentence. Notice also the recognized idiomatic verbal expression *temer o pior* (to fear the worst) as a single node (Galvão et al., 2019b,a).

4. Conclusion

Throughout this article, we have underscored the challenges inherent in implementing standard AMR directives and have explored the potential of the LMR annotation proposal. It is evident that discrepancies arise not only from variations in original versions or translator choices but also from inconsistencies in applying AMR directives (particularly pronounced in translations into Spanish and Brazilian Portuguese). LMR's approach, which anchors directly to the text, offers a promising solution by providing a representation that is closer to the text and less susceptible to the inherent inconsistencies in the process of abstracting the meaning of a text.

In our future endeavors, we intend to expand the annotated texts in LMR, completing the annotation of *O Principezinho* (The Little Prince) and incorporating texts from various genres and domains, including more legal texts.

We plan to develop tools to facilitate faster and more efficient annotation implementation, including: (a) a lemmatizer to associate text forms with lemmas and unique identifiers in the lexicon-grammar; (b) a tool for constructing LMR graphs, which instantiate argument positions of predicative elements and mark positions for anaphora resolution, ensuring formal consistency; (c) a tool for converting graphs into graphical or PENMAN format to facilitate interpretation; (d) a tool for comparing annotations and assessing agreement among annotators, and subsequently, across translations in different languages. With a more extensive corpus, our objective is to develop an LMR parser for automatic representation generation, with the potential for several NLP applications.

5. Acknowledgments

This work was partially supported by Portuguese national funds through Fundação para a Ciência e a Tecnologia FCT (Reference: UIDB/50021/2020, DOI: 10.54499/UIDB/50021/2020) and by European funds through the European Commission (Project: iRead4Skills, Grant number: 1010094837, Topic: HORIZON-CL2-2022-TRANSFORMATIONS-01-07, DOI: 10.3030/101094837).

6. Bibliographical References

Rafael Anchiêta. 2020. *Abstract Meaning Representation Parsing for the Brazilian Portuguese Language*. Ph.D. thesis, Universidade de São Paulo.

Zahra Azin and Gülşen Eryiğit. 2019. Towards Turkish Abstract Meaning Representation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 43–47, Florence, Italy. 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 *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*, pages 178–186.

Jorge Baptista. 2005. *Sintaxe dos Predicados Nominais com* ser de. Fundação para a Ciência e a Tecnologia & Fundação Calouste Gulbenkian, Lisboa.

Jorge Baptista. 2012. ViPEr: A Lexicon-Grammar of European Portuguese Verbs. In *31e Colloque International sur le Lexique et la Grammaire*, pages 10–16.

Jorge Baptista. 2013. Viper: uma base de dados de construções léxico-sintáticas de verbos do português europeu. *Actas do XXVIII Encontro da APL-Textos Selecionados*, pages 111–129.

Jorge Baptista. 2024a. Lexical Meaning Representation - Guidelines. Technical report, University of Algarve/INESC-ID Lisboa. `https://gitlab.hlt.inesc-id.pt/u000803/lmr4pt/`.

Jorge Baptista. 2024b. LMR4PT - Principezinho. Technical report, University of Algarve/INESC-ID Lisboa. `https://gitlab.hlt.inesc-id.pt/u000803/lmr4pt/`.

Jorge Baptista and Rafael Crismán Pérez. 2021. Auxiliary verb constructions in Portuguese and Spanish: a comparative study and its applications as second languages. *Revista de Lenguas Modernas*, 34:39–57.

Jorge Baptista and Nuno Mamede. 2020a. *Dicionário gramatical de verbos do português*. Universidade do Algarve.

Jorge Baptista and Nuno Mamede. 2020b. Syntactic Transformations in Rule-Based Parsing of Support Verb Constructions: Examples from European Portuguese. In *9th Symposium on Languages, Applications and Technologies (SLATE 2020)*, pages 11:1–11:14. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

Jorge Baptista and Nuno Mamede. 2020c. ViPEr v.1. Portulan-CLARIN-PT repository hosted at Research Infrastructure for the Science and Technology of Language, Handle: `https://hdl.handle.net/21.11129/0000-000D-F91E-A`.

Jorge; Baptista, Nuno; Mamede, and Fernando Gomes. 2010. Auxiliary verbs and verbal chains in European Portuguese. In *Computational Processing of the Portuguese Language*, number 6001 in Lecture Notes in Computer Science / Lecture Notes in Artificial Intelligence, pages 110–119, Berlin. PROPOR 2010, Springer.

Claire Bonial, Bianca Badarau, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Tim O'Gorman, Martha Palmer, and Nathan Schneider. 2018. Abstract Meaning Representation of constructions: The more we include, the better the representation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Marco Damonte and Shay B. Cohen. 2018. Cross-lingual Abstract Meaning Representation parsing. 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 1146–1155, New Orleans, Louisiana. Association for Computational Linguistics.

Marco Damonte, Shay B. Cohen, and Giorgio Satta. 2017. An incremental parser for Abstract Meaning Representation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 536–546, Valencia, Spain. Association for Computational Linguistics.

Ana Galvão, Jorge Baptista, and Nuno Mamede. 2019a. New developments on processing European Portuguese verbal idioms. In *12th Symposium in Information and Human Language Technology*, pages 229–238, Salvador, BA (Brazil).

Ana Galvão, Jorge Baptista, and Nuno Mamede. 2019b. Processing European Portuguese Verbal Idioms: From the Lexicon-Grammar to a Rule-based Parser. In *Computational and Corpus-based Phraseology. Proceedings of the Third International Conference EUROPHRAS 2019*, pages 70–77, Malaga (Spain). Tradulex.

Maurice Gross. 1981. Les bases empiriques de la notion de prédicat sémantique. *Langages*, 15(63):7–52.

Zellig Harris. 1964. The elementary transformations. In Henry Hiz, editor, *Papers on Syntax*, pages 211–235. D. Reidel Pub. Co.

Zellig Sabettai Harris. 1976. *Notes du Cours de Syntaxe*. Seuil, Paris. (edited by Maurice Gross).

Zellig Sabettai Harris. 1991. *A Theory of Language and Information. A Mathematical Approach*. Clarendon Press, Oxford.

Eduard Hovy and Julia Lavid. 2010. Towards a 'science'of corpus annotation: a new methodological challenge for corpus linguistics. *International journal of translation*, 22(1):13–36.

Bin Li, Yuan Wen, Weiguang Qu, Lijun Bu, and Nianwen Xue. 2016. Annotating the little prince with Chinese AMRs. In *Proceedings of the 10th Linguistic Annotation Workshop held in conjunction with ACL 2016 (LAW-X 2016)*, pages 7–15, Berlin, Germany. Association for Computational Linguistics.

Ha Linh and Huyen Nguyen. 2019. A case study on meaning representation for Vietnamese. In *Proceedings of the First International Workshop on Designing Meaning Representations*, pages 148–153, Florence, Italy. Association for Computational Linguistics.

Ronaldo Martins. 2012. Le Petit Prince in UNL. In *LREC*, pages 3201–3204. Citeseer.

Christian MIM Matthiessen and John A Bateman. 1991. Text generation and systemic-functional linguistics: experiences from English and Japanese. *Pinter Publishers*.

Noelia Migueles Abraira. 2017. *A Study Towards Spanish Abstract Meaning Representation*. University of the Basque Country. (Master thesis).

Christian Molinier and Françoise Levrier. 2000. *Grammaire des adverbes: description des formes en* -ment. Droz, Genève.

Tim O'Gorman, Michael Regan, Kira Griffitt, Ulf Hermjakob, Kevin Knight, and Martha Palmer. 2018. AMR beyond the sentence: the multi-sentence AMR corpus. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3693–3702, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Elif Oral, Ali Acar, and Gülşen Eryiğit. 2024. Abstract Meaning Representation of Turkish. *Natural Language Engineering*, 30(1):171–200.

James Pustejovsky, Ken Lai, and Nianwen Xue. 2019. Modeling quantification and scope in Abstract Meaning Representations. In *Proceedings of the First International Workshop on Designing Meaning Representations*, pages 28–33, Florence, Italy. Association for Computational Linguistics.

Eloize Seno, Helena Caseli, Marcio Inácio, Rafael Anchiêta, and Renata Ramisch. 2022. XPTA: um parser AMR para o português baseado em uma abordagem entre línguas. *Linguamática*, 14(1):49–68.

Reza Takhshid, Razieh Shojaei, Zahra Azin, and Mohammad Bahrani. 2022. Persian abstract meaning representation.

Hiroshi Uchida, M Zhu, and T Della Senta. 1996. Unl: Universal networking language–an electronic language for communication, understanding, and collaboration. *Tokyo: UNU/IAS/UNL Center*.

Shira Wein and Julia Bonn. 2023. Comparing UMR and cross-lingual adaptations of AMR. In *Proceedings of the Fourth International Workshop on Designing Meaning Representations*, pages 23–33, Nancy, France. Association for Computational Linguistics.

Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, et al. 2013. Ontonotes release 5.0 ldc2013t19. *Linguistic Data Consortium, Philadelphia, PA*, 23:170.

Nianwen Xue, Ondřej Bojar, Jan Hajič, Martha Palmer, Zdeňka Urešová, and Xiuhong Zhang. 2014. Not an interlingua, but close: Comparison of English AMRs to Chinese and Czech. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1765–1772, Reykjavik, Iceland. European Language Resources Association (ELRA).

Adjudicating LLMs as PropBank Annotators

Julia Bonn*[1], Harish Tayyar Madabushi*[2], Jena D. Hwang[3], Claire Bonial[4]
[1]University of Colorado, Boulder,[2] University of Bath, [3]Allen Institute for AI,
[4]Army Research Lab
julia.bonn@colorado.edu

Abstract

We evaluate the ability of large language models (LLMs) to provide PropBank semantic role label annotations across different realizations of the same verbs in transitive, intransitive, and middle voice constructions. In order to assess the meta-linguistic capabilities of LLMs as well as their ability to glean such capabilities through in-context learning, we evaluate the models in a zero-shot setting, in a setting where it is given three examples of another verb used in transitive, intransitive, and middle voice constructions, and finally in a setting where it is given the examples as well as the correct sense and roleset information. We find that zero-shot knowledge of PropBank annotation is almost nonexistent. The largest model evaluated, GPT-4, achieves the best performance in the setting where it is given both examples and the correct roleset in the prompt, demonstrating that larger models can ascertain some meta-linguistic capabilities through in-context learning. However, even in this setting, which is simpler than the task of a human in PropBank annotation, the model achieves only 48% accuracy in marking numbered arguments correctly. To ensure transparency and reproducibility, we publicly release our dataset and model responses.

Keywords: PropBank, Semantic Role Labeling, LLM Evaluation

1. Introduction

The increasing generative power of LLMs presents ample opportunity for NLP resource practitioners to employ it for large-scale annotation efforts, which have traditionally been costly and labor intensive. Various studies have touted the promises of these large scale language models' capabilities for syntactic and semantic analyses (c.f. Tan et al. 2024; Savelka and Ashley 2023; Shin and Van Durme 2022). Other works suggest that they are still yet to achieve the type of capabilities that are needed to make them truly useful in language resource building capacities (Lu et al., 2023; Ettinger et al., 2023; Bonial and Tayyar Madabushi, 2024). In this work,[1] we empirically test the feasibility for using state-of-the-art LLMs for conducting large scale linguistic annotation, using PropBank as a test bed. More concretely, we ask, do GPT-3.5 and GPT-4, which excel in language generative capabilities, possess the ability to produce viable PropBank annotation?

Our choice of PropBank annotation as the testbed is motivated by the ways in which PropBank annotation is rooted in both syntax and semantics. Although the task of PropBank is primarily semantic role labeling, the semantic roles assigned depend upon the choice of a given relation's coarse-grained sense. Sense distinctions in PropBank were made based upon differences in semantic roles as well as syntactic behaviors—namely the subcategorization frame of a relation or the ways in which the semantic arguments are realized syntactically (e.g., as subjects, direct objects, or obliques). Thus, PropBank senses or "rolesets" reflect a set of semantic roles that are realized in a syntactically distinct way. As a result, PropBank is a powerful resource that provides explicit mappings between particular syntactic patterns of argument expression and the semantic roles of those arguments, enabling a shallow semantic analysis facilitated by clearly recognizable syntactic patterns. Given that LLMs have been touted for their abilities with respect to both syntax and semantics, we seek to test whether the mapping of syntactic constituents to particular semantic roles can be accomplished by LLMs.

The primary contribution of this research is an initial assessment of the meta-linguistic capabilities of LLMs, where we design three prompts meant to dissect LLMs' abilities with respect to the PropBank tasks of both argument annotation and sense or roleset annotation. We test LLMs' ability to accomplish roughly the equivalent task as human PropBank annotation via few-shot in-context prompting. We also test two additional settings: a more difficult setting testing LLMs' zero-shot knowledge of PropBank (no in-context examples); and an easier setting, where the LLM is provided with not only the few-shot examples but also the correct roleset with expected roles.

Our findings show that even GPT-4 (best model)

*Equal contribution

[1] Dataset and model responses available at https://github.com/H-TayyarMadabushi/Adjudicating-LLMs-as-PropBank-Annotators

		Match Types		
Model	**Setting**	**Exact**	**Core-Arg**	**Num-Arg**
GPT-3.5	0-shot	8.6%	8.6%	17.1%
	3-shot	11.4%	17.1%	37.1%
	3-shot+rs	2.9%	2.9%	20.0%
GPT-4	0-shot	8.6%	17.1%	34.3%
	3-shot	14.3%	20.0%	42.9%
	3-shot+rs	**22.9%**	**22.9%**	**48.6%**

Table 1: We report on positive matches for GPT-3.5 and GPT-4 over three prompt settings: 0-shot, 3-shot, and 3-shot with roleset (3-shot+rs).

		Match Types		
Construction	**N**	**Exact**	**Core-Arg**	**Num-Arg**
Transitive	14	50.0%	50.0%	85.7%
Intransitive	13	7.7%	7.7%	23.1%
Middle	8	12.5%	12.5%	25.0%

Table 2: We report the percentage of positive matches for our **best-performing prompt and model** combination: GPT-4 with the few-shot prompt that includes the correct roleset. N refers to the number of instances available for each construction.

generally struggles to assign correct semantic roles to the arguments across syntactic realizations, achieving 42.9% accuracy in the few-shot setting (Table 1). When the roleset is predefined alongside examples, the model performance does improve to 48.6%; however, that is abysmally low in comparison to the reported PropBank human average of 88.3%. As expected, the zero-shot setting is the most difficult for the LLMs (34.3%).

Furthermore, we show that GPT-4's relatively poor performance stems from its apparent inability to generalize semantics across the various syntactic realizations. The highest successes are attributed to the transitive construction (best 85.7%) where syntax maps canonically to PropBank's argument numbering (i.e. Arg0-5). For intransitive and middle voice constructions, performance drops considerably (best 25.0%).

2. Background & Motivation

2.1. PropBank Annotation

Born in the early 2000s, The Proposition Bank (PropBank) changed the world of lexical semantics in NLP by using syntactic parses as a scaffolding for the much more difficult problem of parsing meaning. The underlying idea was that English verbs exhibit patterns in the way they structure their participants both syntactically and semantically, and so by tagging syntactic arguments of a verb with semantic role labels, a system could be trained to understand fundamental propositional semantics (i.e. who did what to whom, when and how?) using syntactic cues (Palmer et al., 2005).

PropBank's main innovation was in creating a large scale inventory of rolesets (sense disambiguated predicate argument structures) for English verbs, and then having expert human annotators apply them to syntactic parse trees from the Penn TreeBank (Taylor et al., 2003). The PropBank roleset lexicon consists of verb lemmas organized into frame files. Each frame file contains one or more rolesets representing the different semantic senses associated with the verb, with each roleset providing a predicate label, a written sense definition, and a list of roles corresponding to the semantically-essential participants of the event. PropBank roles are numbered and given short written descriptions rather than more traditional thematic role labels as a way of splitting the difference between semantic and syntactic primacy of the argument. For example, Arg0s correspond to proto-agents (Dowty, 1991), which also tend to occur as syntactic subjects on verbs, and Arg1s generally correspond to proto-patients, which often occur as syntactic objects. Consider, for example, the following rolesets for the verb *deal*:[2]

Verb: deal
Roleset: deal.01 (*handle, deal with, transaction*)
ARG0: dealer (or all dealers)
ARG1: co-dealer
ARG2: subject/type of transaction
ARG3: value of transaction

Roleset: deal.02 (*play cards, distribute something*)
ARG0: distributor
ARG1: cards, thing distributed
ARG2: other player(s), distributed to

The annotation schema itself was relatively simple. For every instance of a verbal relation in a corpus sentence, annotators would first select a roleset and then tag the nodes in the parse tree governed by the verb with either a) a numbered argument from the roleset, or b) one of a small inventory of general semantic modifier args (ArgMs, e.g., ArgM-LOC (location), ArgM-DIS (discourse markers), ArgM-MNR (manners and instruments)) (Bonial et al., 2010). Annotators were presented all of the instances of a given verb lemma from the corpus as a single task, and were able to see all of the rolesets associated with that lemma in a dropdown menu (Choi et al., 2010). For each roleset, they were able to see the definition, the roles with their descriptions, and they were able to

[2] All rolesets provided in this paper are copied directly without changes from `https://propbank.github.io/v3.4.0/frames/`.

open a window that showed a variety of annotated example sentences.

One of PropBank's greatest successes was that, across a wide range of corpora and domains, human annotators were able to make these judgments easily and consistently. Inter-annotator agreement (IAA) was consistently high for PropBank—Bonial et al. (2017) report "exact match" (all constituents and arguments match precisely) IAA for English verbal relations at 84.8%, and "core-arg match" (numbered arguments match and ArgMs match, but the specific ArgM, such as Temporal or Locative, need not match) of 88.3%.

2.2. Related Works & Motivation

The benefits of being able to produce annotations with little training data has become an alluring prospect for resource practitioners in NLP. In the recent years, LLMs have been used to collect large-scale datasets (c.f. Liu et al. 2022; Shin et al. 2020) or to distill data to enable smaller models (c.f. Bhagavatula et al. 2022; West et al. 2021) as a means of reducing the cost burden that large-scale annotation efforts may incur. These achievements have been made possible by LLMs' capability to produce impressive generations, which have been attributed to an emergent capability to do semantic reasoning (Srivastava et al., 2023; Wei et al., 2022).

Recently, however, several works have cast scrutiny over the LLM capabilities for grasping semantic components of language and for targeted semantic analysis. Lu et al. (2023) have suggested that ability to tackle complex tasks is not necessarily emergent. Rather, models are adept at leveraging in-context learning to tackle complex tasks.[3] To refine our understanding and better delineate the parameters necessary to prompt LLMs to exhibit complex analytical abilities, we undertake experiments employing prompts both with and without illustrative examples. These experiments aim to establish the optimal prompt format conducive to eliciting LLM abilities that enable us to solve meta-linguistic tasks such as this, while also serving as a method for exploring the capabilities of and limitations of LLMs.

In terms of the level of semantic analysis LLMs are able to accomplish, some research shows that larger LLMs are able to sort sentences by semantic similarity based on constructional semantics (e.g., grouping together *She blinked the tears off of her eyelashes* and *She wiped the flour off of the table*), while smaller LLMs are only able to sort sentences by lexical semantics (e.g., grouping together *blink, cough, breathe* regardless of their broader constructional setting) (Li et al., 2022). However, recent research suggests that even the largest models (GPT-4) are unable to recognize the semantic similarity of events expressed in argument structure constructions (Goldberg, 2003), such as the resultative, when non-canonical verbs are found in these constructions (e.g., *He yelled himself hoarse* as opposed to *He made himself hoarse by yelling*) (Bonial and Tayyar Madabushi, 2024). Even if LLMs are able to group some sentences by semantic and constructional similarity, there is evidence suggesting that the models are not able to infer the appropriate semantics from constructions such as *The more I study it, the less I understand it* (Weissweiler et al., 2022).

Wilson et al. (2023) evaluate the extent to which models in the BERT family are able to generalize different types of linguistic knowledge, including what they call "Type 2 knowledge," which allows speakers to predict word occurrences in new, structurally related contexts they have not explicitly encountered before, based on their understanding of how thematic roles are typically assigned across different grammatical structures. The authors use fine-tuning and introduce novel tokens in a fixed structural context to evaluate the extent to which pre-trained language models generalize to Type 2 knowledge. The authors find that PLMs can generalize to Type 2 knowledge only to a very small extent, and do not generalize across active and passive sentences. While these results are certainly relevant to our own research question, we emphasize that we are testing much larger models, where research has suggested distinct potential for in-context learning (Wei et al., 2023).

Moreover, Ettinger et al. (2023) have shown that LLMs can readily achieve surface level semantic analysis such as locating the main predicate and its core arguments (i.e. retrieving the "who-did-what-to-whom"). However, when tasked to capture a more complex semantic analysis as required by the structured AMR framework, the models fail miserably even when presented with a diverse set of in-context examples. Thus, in this work we turn to PropBank, which provides a relatively simple semantic annotation framework revolving around identifying the who-did-what-to-whom information of a verb, which may be a more reasonable level of semantic decomposition for LLMs to grasp.

However, despite the simplicity of the PropBank framework, we also recognize the annotation demands a level of comprehension beyond that of mere pattern recognition. It requires the comprehension of the elements of the sentence and their associated forms. Thus, we hypothesize that the effectiveness of LLMs on this task is likely to be limited, especially due to the complexity of this task which requires a certain "understanding" of the meaning of sentences. This is especially likely

[3] In-context learning refers to the capability of LLMs to perform tasks based on minimal examples.

given the propensity of LLMs to generate linguistically fluent, but factually or logically inconsistent sentences (Rawte et al., 2023).

In light of the evolving discourse surrounding LLMs and their capabilities, this work aims to explore the utility of LLMs in generating PropBank annotations. Specifically, we aim to answer the following research questions: a) How effective are LLMs at generating PropBank annotations, and b) What is the most effective way of prompting LLMs for the purpose of PropBank annotation?

3. Evaluation Framework

3.1. Verb and Construction Targets

To capture a wide variety of semantic and syntactic realizations, we select 7 verbs from 7 distinct VerbNet Classes (Schuler, 2005) for the evaluation and analysis of LLM capability for PropBank annotation. The verbs are listed in Table 3. While PropBank annotations are inclusive of both verb and non-verbal relations (e.g., `pitch.04` serves both *the White House* ***pitch*** and *the proposal* ***pitched*** *by the White House*), for the purposes of this work, we specifically focus only on verbal relations.

These verbs are selected on the basis of their ability to participate in three syntactic realizations (henceforth, constructions): transitive, intransitive, and middle voice. These constructions map semantic arguments to their syntactic element quite distinctly. As such, they allow us to evaluate if LLMs can appropriately assign what are generally Arg0 prototypical agents and Arg1 prototypical patients to the correct arguments, despite these fundamental constructional differences. For example, in intransitive realizations, the subjects may be animate Agents or Causes (e.g., *John writes well*), but we may also see inanimate Patients undergoing a change of state (e.g., *The chair broke*). In the middle voice, it is the Theme or Patient that sits in the subject position with the Agent unmentioned (e.g., *the cards deal smoothly*). Further details on the data collection and distinction between intransitive and middle voice can be found in Appendix A. Thus, each evaluation instance requires the model to cue on both the syntactic and lexical semantic information to determine whether it is Arg0 or Arg1 that likely sits in the subject position. From PropBank IAA, we know that human annotators can easily track these alternations. In this work, we investigate whether the models can do so as well.

3.2. Evaluation Set and Data Source

For compiling our exploratory corpus for evaluating LLMs, we leverage the Corpus of Contemporary American English (COCA) (Davies, 2008), which enables targeted search for particular verb in the syntactic realizations of our interest. As COCA does not furnish PropBank annotations, the extracted sentences are annotated for verbal relation targets by three of the authors previously trained extensively in PropBank annotation standards.

From COCA, we extract sentences for each of the 7 verbs with 5 usages per verb (aiming for 2 transitive, 2 intransitive, and 1 middle voice construction) resulting in a total of 35 sentences in the evaluation set. Additionally we extract 3 instances corresponding to the three constructions for in-context examples used in our few-shot setting. Further details are included in Appendix A.

The purpose of this annotated dataset is an initial exploration of LLM capabilities; it is not large enough to serve as a full diagnostic evaluation set. Although we considered leveraging some of the existing PropBank corpus annotations, we opted to annotate new sentences not included in any past PropBank release to avoid the possibility that the model's training data included the existing annotated corpora.

3.3. Models & Prompting Strategies

The capabilities of LLMs are inherently determined by the extent of their training and the scale of their parameters. As such, in assessing the proficiency of LLMs as effective PropBank annotators, our analysis centers on two prominent and powerful language models, GPT-3.5-turbo-0301 and GPT-4-0613. The experiments are conducted via the OpenAI API using a temperature setting of 0. A temperature of 0 is chosen to enforce deterministic output generation, wherein the models select the most probable next token thus ensuring reproducibility. Due to the deterministic nature of our experiments, we run each of them once.

The choice of the specific prompts employed when interfacing with LLMs has been identified as a critical factor influencing their performance. We employ the following three prompting formats:

- **0-shot setting**: The model is instructed to annotate the provided sentence using PropBank annotations. This is a setting we expect to be harder than human PropBank annotation as no examples nor rolesets are made available.

- **3-shot setting**: The model is provided with 3 examples in a setting that is roughly equivalent to a human PropBank annotation set up—examples are given and, in addition to completing annotation, the annotator must decide on the roleset.

- **3-shot roleset setting** (3-shot+rs): Along with the examples, the model is provided with the roleset associated with the input sentence. This setting is easier than human annotation—examples and the rolesets with expected roles are provided.

Verb	VerbNet Class	Corpus Example (corresponding construction in parenthesis)
break	Break-45.1	I think you were badly cut when the chair broke under you. (intransitive)
pour	Pour-9.5	The beer pours a hazy yellow color with a huge white head. (middle)
write	Say-37.7-1	It was due to illness and the doctor wrote a letter saying I couldn't fly. (transitive)
deal	Give-13.1	I saw that dude dealing drugs. (transitive)
smell	See-30.1	Butterflies smell with their feet. (intransitive)
parse	No VN Entry	Spivak is the most gender-free pronoun that parses well in English. . . (middle)
rain	Weather-57	In spring and fall it rains occasionally. (intransitive)
hike	Run-51.3.2	This trail hikes through a portion of the historic area...(middle)

Table 3: We focus on 7 verbal relations for evaluation set with 5 usages for each verb for a total of 35 sentences. We use the 8th verbal relation ("hike") for in-context examples for prompting.

In addition to the specific prompt format, the exact wording of the prompt itself has been found to have an effect on the output generated by LLMs. Given the inexact nature of prompt engineering, we conduct preliminary tests focused on subjective assessments of output variations on a small number of test samples. While there could always be a more effective prompt, identifying such an optimal prompt is not straightforward. Additionally, our aim is to assess how annotators typically would interact with LLMs.

In human annotation, examples provided during annotation do not necessarily use the same verb or voice as the sentence being annotated. Thus, in selecting examples, we always use the same (static) set of examples, involving the verb *hike*, which differ from the evaluation dataset.[4] We conducted experiments using a range of prompts aimed at identifying the most effective wording and format, using a small subset of our data. The final prompt we use is shown Appendix B.

We also note that providing in-context examples allows us to evaluate models that may not have been explicitly trained with PropBank annotations. By incorporating in-context examples, we circumvent the need for models to undergo specific fine-tuning (also called instructional fine-tuning) for understanding instructions pertaining to PropBank.

3.4. Metrics for Evaluation

We use three evaluation metrics that mirror evaluation metrics used to report human IAA for PropBank annotation (see Albright et al. (2013) for a summary of metrics). Exact match represents the strictest match, while the rest are more relaxed measures.

- **Exact Match:** LLM annotation matches the manually produced, gold standard annotation with respect to constituent boundaries as well as the same role number or the same ArgM type identified for each phrases.
- **Core-Arg Match:** LLM's constituent boundaries match and have the same numbered roles labeled as the human annotation. ArgMs also match in terms of *being* argMs, although the distinctions between the individual ArgM labels is ignored. This relaxed measure allows for ArgM type differences as observed in human annotation. For example, *The paper presented* ***at a 2020 ACL*** could plausibly be marked as either ArgM-TMP or ArgM-LOC.
- **Number-Arg Match:** LLM and human annotations are matched with respect to the heads of argument phrases (correct participant is identified, ignoring precise constituent boundaries), and with respect to numbered arguments only, ignoring ArgM annotations. Here, we are are primarily interested in the correct assignment of Arg0 and Arg1 despite syntactic differences in their realization or their omission.[5]

4. Results

Here we report results for both GPT-3.5 performance and GPT-4 performance across our three different prompt settings: zero-shot, 3-shot without the roleset information given, and 3-shot with the correct roleset given in the prompt. In Table 1, we report the percentage of positive matches across each of the match types from strictest to loosest: Exact, Core-Arg, and Numbered-Arg match. In the sections to follow, we discuss and provide match

[4] While the *hike* examples are provided in transitive, intransitive, and middle constructions, we acknowledge that there may be an effect of using a single verb across the few-shot examples. We provide a follow-on experiment in Section 5 that examines results where the prompt verb and voice match the test usage.

[5] The roleset specification in the 3-shot+rs setting includes the description of the core arguments only, without reference to the various ArgMs that PropBank allows. Thus, outside of the ArgMs included in examples in the few-shot setting, the model is given no guidance on ArgM annotation, whereas human PropBank annotators would be trained to identify ArgM types. Number-Arg Match is designed to assess the generation without unfairly penalizing the model for mistakes in ArgM.

and error examples for each prompt setting and finally for the different sentence types (transitive, intransitive, middle voice).

4.1. Zero-Shot Setting

In the zero-shot setting, we prompt the model, "Given the following verb and sentence, produce a PropBank annotation of the verb sense and its arguments." In this setting, we provide only the target verb and sentence, we do not provide potential rolesets or the correct roleset. With this prompt, GPT-3.5 is only able to provide exact and core-arg matches for two relatively straightforward transitive sentences:

1. (This reporter)-ARG0 smells-REL (another Emmy)-ARG1[6]
2. (A fringe of activists)-ARG0 broke-REL (some doors and windows of the halls)-ARG1 and committed two minor assaults.

The numbered-arg matches that GPT-3.5 is able to obtain are also largely (5 of 7 matches) of the transitive type.

GPT-4 performs much better than GPT-3.5 in the zero shot setting. GPT-4 matches on the same transitive sentences that GPT-3.5 was able to match in this setting, and it is also able to provide core-arg matches for 2 intransitives and 2 middle voice usages, including, for example:

3. GPT-4 Annotation: Use the heavy floss because (the fine floss)-ARG1 breaks-REL (easily)-ARGM-ADVERBIAL
4. GPT-4 Annotation: (This fellow)-ARG0 writes-REL (abominably)-ARGM-ADVERBIAL

Note that the above sentences are only core-arg matches, as opposed to exact matches, due to differences in the specific ArgMs marked. The gold standard marks what was annotated by GPT-4 as ArgM-Adverbial instead as ArgM-Manner. Interestingly, as we describe in the next section, GPT-4 is not able to correctly annotate the above sentences in the few-shot setting.

4.2. Few-Shot Setting

In the few-shot setting, we prompt the model in the same way, but we also provide three example annotations that all use the verb *hike*, exemplified in transitive, intransitive, and middle voice constructions. We then provide the target verb and sentence. We do not provide any information regarding the relevant roleset. Thus, this setting is very similar to what a human annotator would face, as they would not have seen the particular target annotation instance before, though they may have seen variety of other PropBank annotation examples during their training. Note that some generalization is required in moving from the examples of a different verb and the alternations in Arg0 and Arg1 seen for that verb, and the parallel syntactic alterations for the target verb.

While both GPT-3.5 and GPT-4 show improvement in this setting, the improvement is not as straightforward as one might expect. Specifically, the gains are made primarily with respect to superior annotation of transitive usages. For example, GPT-4 in particular fails to match on the middle and intransitive sentences (3) and (4) above by shifting the Arg1, *the fine floss*, to an Arg0, while also shifting the manner adjunct, *abominably* to an Arg1. We hypothesize therefore that adding the examples for comparison causes the model to overgeneralize where numbered arguments should be used, and specifically where Arg0 should be used, perhaps given that most of the *hike* examples involve an Arg0 subject.

4.3. Simplified Annotation Task in Few-Shot Setting

In the final prompt setting we provide the most information, simplifying the annotation task by including the correct roleset in the prompt. Thus, in addition to examples of how argument numbers are applied across the three constructions from the verb *hike*, we also describe explicitly how the argument numbers map to the semantic roles, expressed in natural language (as opposed to traditional thematic role labels), for the target verb.

We find that GPT-3.5 performs worse in this setting, with the numbered-arg matches falling from 37.1% in the few-shot setting to 20.0% when we now provide the roleset. When we examine where new errors were introduced in this setting, we find that example (2), which was consistently annotated correctly in the zero-shot and few-shot (without the roleset) settings, is no longer annotated correctly. Instead, the model over-extends the application of the numbered arguments specified in the roleset (see Figure 1), which was provided to the model .

5. (A fringe of activists)-ARG0 broke-REL (some doors and windows of the halls)-ARG1 (away from the halls)-ARG4

Note that the phrase GPT-3.5 assigns as the Arg4 (thing broken away from) is not present in the original sentence. The model adds this to the annotation despite explicit prompting to only use the words found in the sentence. Similarly, the inclusion of the roleset seems to have entirely derailed GPT-3.5's annotation, resulting in particularly

[6] We use this notation to express the gold standard annotation, we did not expect or require the LLMs to output in this format.

Verb: break
Roleset: break.01 (break, cause to not be whole)
ARG0: breaker
ARG1: thing broken
ARG2: instrument
ARG3: pieces
ARG4: arg1 broken away from what?

Verb: smell
Roleset: smell.02 (emit an odor)
ARG1: stinky thing
ARG2: attribute of arg1

Figure 1: PropBank rolesets *break.01* and *smell.02*

Verb: rain
Roleset: rain.01 (rain)
ARG0: metaphorical agent
ARG1: metaphorical rain
ARG2: rained upon

Figure 2: PropBank rolesets *rain.01*

widespread (and incorrect) application of the numbered arguments:

6. GOLD annotation: I think you were badly cut when (the chair)-Arg1 broke-Rel (under you)-ArgM-Location

7. GPT-3.5 annotation: I think you were (badly)-Arg3 cut when (the chair)-Arg0 (broke)-Rel under (you)-Arg1

GPT-4, in contrast, achieves the best performance in this setting for all match types, with a best result of 48.6% numbered-arg matches overall. Notably, most of this improvement comes in adding matches for the intransitive and middle voice usages, for example, achieving an exact match (whereas no other settings produced any type of match) on this usage of *smell* (see Figure 1).

8. (Our guy)-Arg1 smells-Rel (incredible)-Arg2

Thus, we hypothesize that when the mapping from the roleset to the usage in question is particularly simple and clear, the model is able to precisely apply the roleset information. However, we acknowledge that it cannot handle cases beyond the simple with much success.

4.4. Constituent Matching

A key difference between our prompt setup and the information presented to human annotators is that humans are asked to place the PropBank argument labels on top of Penn TreeBank constituency parses (Marcus et al., 1994). The annotators are instructed place labels only on constituents that are sisters to the verb phrase (i.e. the subject) and sisters of the verb (i.e. the direct object) (Bonial et al., 2010), which is enforced by the PropBank annotation tool (Choi et al., 2010).[7] Pradhan et al. (2022) attribute some of the high IAA to the fact that the placement of annotations is clearly constrained by the syntactic tree.

In our prompting experiments, we do not provide the syntactic tree corresponding to the sentence. Thus, in this section we explore the extent to which our best-performing model, GPT-4, is able to provide constituent matches with the gold standard. A constituent match is based solely on what phrases are treated as annotated arguments, where the argument labels themselves are entirely ignored. We find that in the zero-shot setting, GPT-4 obtains positive constituent matches in 42.9% of the annotations. In the 3-shot setting where no roleset is given, constituent matches are made for 51.4% of the sentences. Finally, in the 3-shot setting where the roleset is given, constituent matches drop slightly to 48.6%. The fact that constituent matches are hovering around 50% is a trend that suggests that constituent matching is likely a large source of annotation error.

To gain a sense of what the constituent mismatches look like, consider the following example, given the following roleset for *rain* (Figure 2):

9. GOLD annotation: On days (when)-ArgM-Temporal (it)-Arg0 rains-Rel (nonstop)-ArgM-Temporal, they throw sheets of plastic over their hung wash.

10. GPT-4 annotation: (On days when)-ArgM-Temporal (it)-Arg0 rains-Rel (nonstop)-ArgM-Adverbial, they throw sheets of plastic over (their hung wash)-Arg2

Note that *their hung wash* is what might have been rained upon, had they not thrown sheets of plastic over it. Thus, while there may be some plausible justification for calling this Arg2, it is not in a syntactic position to be considered a PropBank argument for *rain*.

The numbered-arg match type does not require constituent matches, but instead asks if the numbered arguments are assigned correctly to phrases with the same head. Thus, there are instances in our data where the constituents annotated do not match, but the annotation is assigned a numbered-arg match. Generally, these are cases where the model fails to annotate an adjunct argument altogether, or when constituent boundaries are slightly off; for example, consider the following case

[7] This training follows generative assumptions that the verbal relation assigns theta roles to its arguments, and that its arguments appear in these positions and only these positions.

of numbered-arg match that is not a constituent boundary match:

11. GOLD annotation: (The Palestinians)-ARG0 rained-REL (stones)-ARG1 (down)-ARGM-DIRECTION (onto Jews praying at the Western Wall below)-ARG2, (injuring 11)-ARGM-ADVERBIAL

12. GPT-4 annotation: (The Palestinians)-ARG0 rained-REL (stones)-ARG1 down (onto Jews praying at the Western Wall below, injuring 11)-ARG2

4.5. Trends Across Transitive, Intransitive, Middle

A key research question in this evaluation is whether or not LLMs can act as PropBank annotators, where the most critical aspect of the annotation is correctly assigning argument numbers across different syntactic realizations of the same relation. Thus, in this section, we focus on performance across transitive, intransitive, and middle voice constructions. Note that our evaluation includes the same 7 verbs exhibited in each of these construction types, and the few-shot examples are also one of each construction. For this analysis, we focus on our best-performing model and prompt combination—GPT-4 in the 3-shot setting with the correct roleset provided.

As we can observe in Table 2, the model achieves by far the most matches (85.7% numbered-arg matches) for transitive usages. The model can only achieve the most relaxed measure, numbered-arg match, about 25% of the time across intransitive usages (23.1 %) and middle voice usages (25.0%). Again, our dataset is small, but from this trend, we conclude that even at its best performance, GPT-4 cannot identify the same semantic roles arising in distinct syntactic realizations. Overall, we see that even for transitives, the best performing model and prompt combination achieves a core-arg match of only 50.0%. We contrast this with the human IAA reported in Bonial et al. (2017), where people achieve an exact match IAA of 84.8% for verbal relations and a core-arg match IAA of 88.3%—and those agreement rates are for verbal relations realized in a wide variety of syntactic realizations.

5. Discussion & Follow-On Experimentation

We started out our study by asking two questions regarding LLM capabilitily with respect to its (in-) ability for to perform PropBank annotation: (a) How effective are LLMs at this task, and (b) What is the most effective way of prompting LLMs for this task. Based on our results, we observe that there is little evidence of any zero-shot meta-linguistic knowledge enabling PropBank annotation. There is some evidence that the larger model can do better with more information—in-context learning is certainly required for the ability to do PropBank annotation. Specifically, we conclude that LLMs are *not* a good replacement for expert linguistic annotators in generating PropBank annotations, and the use of in-context examples is helpful in better guiding LLMs towards the kind of annotations that are more accurate.

To further validate this conclusion, we conducted additional in-context experiments: Concretely, we assessed the models' ability to correctly perform PropBank annotation when in-context examples have the same verb and voice as the target usage to be annotated. This enabled us to gauge the model's capability in a scenario with minimal variation between the in-context example and the model's requirements. Our findings consistently demonstrate that both GPT-3.5 and GPT-4 perform better on this version of the task than on the original one. In fact, we observed that providing explicit information related to the roleset helps models correctly complete the task in instances where they previously do not. Overall, these results indicate that models seem to be effective in following explicit instructions in the form of templated in-context examples, as opposed to being able to generalize from generic instructions akin to those presented to humans.

Importantly, this indicates that resources such as PropBank continue to be useful and, indeed, essential despite the effectiveness of LLMs, regardless of their size. Not only are these datasets helpful in probing the capabilities and limitations of LLMs, they are also likely to be useful in augmenting LLMs with additional capabilities including, for example, a sample-efficient and nuanced interpretation of input sentences.

6. Conclusion and Future Work

Our research indicates that while LLMs may excel at producing natural language text, they also show astonishingly poor capabilities to generalize semantically, especially when it comes to the capacity to produce meta-linguistic annotations that adhere to the annotation standards of the PropBank framework. However, we also show the utility of in-context examples and positive effect of carefully designed prompts in producing better LLM meta-linguistic generations. As PropBank and other linguistic resources remain valuable for semantic analysis, our work suggests that continued research and investment is needed in exploring how to best support in-context learning of meta-linguistic knowledge.

It's worth underscoring that the goal of this study was to assess current model capabilities to do basic meta-linguistic annotation, rather than developing methods by which we can empower models to do PropBank annotation. Our finding that models fail to perform even for manually-selected prototypical constructions with sufficiently clear prompts indicates a failure in meta-linguistic capabilities. Future works to expand on evaluation dataset size will be required to reveal the prevalence of this problem and further explorations with prompt engineering would be necessary to assess the depth of brittleness of model capabilities.

Thus, an immediate future work includes the expansion of the evaluation dataset. While the small size of this dataset is appropriate for the present work that is aimed at an initial exploration of LLM capabitilies, the expansion of this dataset would be necessary to scale up to a full diagnostic set for evaluating models. We expect that a larger evaluation set will be helpful to discover further insights, giving us the capability to make more robust generalizations with regard to model capabilities. Also, the present work was limited to GPT-3.5 and GPT-4. Future directions include expanding the evaluation over other models of varying scale and attested capabilities.

In this work, we have specifically focused on the inclusion of 3-shot and roleset information for prompting experiments. Future studies include an expansion on the prompting types and varieties to better assess and categorize the errors observed in models with the goal of providing more insightful recommendations for meta-linguistic prompting for PropBank annotation.

A broader application of this work is the possibility of leveraging LLMs for building up semantic resources for lower-resource languages with limited capacity for mass-annotation efforts like crowdsourcing. It is yet unclear what the extent of **multilingual** meta-linguistic capabilities of LLMs are. However, a wider net of experiments that include verb-argument behavior different from that of English is a compelling future direction of this research.

7. Ethical Considerations and Limitations

Dataset Size. The goal of the work was to take pulse of LLM capabilities regarding PropBank annotation for the purpose of a close-up manual analysis of the successes and mistakes the LLMs make in the annotation process. For this purpose, the size of the dataset was suitable. However, because the dataset used in this work is indeed very small, we do not recommend the set to be used as a full diagnostic evaluation set.

English Centricity. PropBank is available not only for English, but a wide number of languages and domains. PropBank lexicons and/or corpora now exist for for Chinese (Xue, 2006), Korean (Palmer et al., 2006), Arabic (Zaghouani et al., 2010), Hindi (Vaidya et al., 2013), Portuguese (Duran and Aluísio, 2012), Finnish (Haverinen et al., 2014), and Turkish (Şahin and Adalı, 2018), just to mention those we know well. This work, however, focuses on the aspects of PropBank annotation that is relevant to English only. The findings we offer may not hold for other languages.

8. Bibliographical References

Daniel Albright, Arrick Lanfranchi, Anwen Fredriksen, William F Styler IV, Colin Warner, Jena D Hwang, Jinho D Choi, Dmitriy Dligach, Rodney D Nielsen, James Martin, et al. 2013. Towards comprehensive syntactic and semantic annotations of the clinical narrative. *Journal of the American Medical Informatics Association*, 20(5):922–930.

Chandra Bhagavatula, Jena D Hwang, Doug Downey, Ronan Le Bras, Ximing Lu, Lianhui Qin, Keisuke Sakaguchi, Swabha Swayamdipta, Peter West, and Yejin Choi. 2022. I2d2: Inductive knowledge distillation with neurologic and self-imitation. *arXiv preprint arXiv:2212.09246*.

Claire Bonial, Olga Babko-Malaya, Jinho D Choi, Jena Hwang, and Martha Palmer. 2010. Propbank annotation guidelines. *Center for Computational Language and Education Research Institute of Cognitive Science University of Colorado at Boulder*.

Claire Bonial, Kathryn Conger, Jena D Hwang, Aous Mansouri, Yahya Aseri, Julia Bonn, Timothy O'Gorman, and Martha Palmer. 2017. Current directions in english and arabic propbank. *Handbook of linguistic annotation*, pages 737–769.

Claire Bonial and Harish Tayyar Madabushi. 2024. A construction grammar corpus of varying schematicity: A dataset for the evaluation of abstractions in language models. In *Proceedings of the Language Resources and Evaluation Conference (LREC)*.

Jinho D Choi, Claire Bonial, and Martha Palmer. 2010. Propbank instance annotation guidelines using a dedicated editor, jubilee. In *LREC*. Citeseer.

Mark Davies. 2008. The corpus of contemporary american english (coca): 560 million words, 1990-present.

David Dowty. 1991. Thematic proto-roles and argument selection. *language*, 67(3):547–619.

Magali Sanches Duran and Sandra Maria Aluísio. 2012. Propbank-br: a brazilian treebank annotated with semantic role labels. In *LREC*, pages 1862–1867.

Allyson Ettinger, Jena Hwang, Valentina Pyatkin, Chandra Bhagavatula, and Yejin Choi. 2023. "you are an expert linguistic annotator": Limits of LLMs as analyzers of Abstract Meaning Representation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8250–8263, Singapore. Association for Computational Linguistics.

Adele E Goldberg. 2003. Constructions: A new theoretical approach to language. *Trends in cognitive sciences*, 7(5):219–224.

Katri Haverinen, Jenna Nyblom, Timo Viljanen, Veronika Laippala, Samuel Kohonen, Anna Missilä, Stina Ojala, Tapio Salakoski, and Filip Ginter. 2014. Building the essential resources for finnish: the turku dependency treebank. *Language Resources and Evaluation*, 48:493–531.

Bai Li, Zining Zhu, Guillaume Thomas, Frank Rudzicz, and Yang Xu. 2022. Neural reality of argument structure constructions. *arXiv preprint arXiv:2202.12246*.

Alisa Liu, Swabha Swayamdipta, Noah A. Smith, and Yejin Choi. 2022. WANLI: Worker and AI collaboration for natural language inference dataset creation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6826–6847, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Sheng Lu, Irina Bigoulaeva, Rachneet Sachdeva, Harish Tayyar Madabushi, and Iryna Gurevych. 2023. Are emergent abilities in large language models just in-context learning?

Mitch Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The penn treebank: Annotating predicate argument structure. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1):71–106.

Martha Palmer, Shijong Ryu, Jinyoung Choi, Sinwon Yoon, and Yeongmi Jeon. 2006. Korean propbank. *LDC Catalog No.: LDC2006T03 ISBN*, pages 1–58563.

Sameer Pradhan, Julia Bonn, Skatje Myers, Kathryn Conger, Tim O'gorman, James Gung, Kristin Wright-Bettner, and Martha Palmer. 2022. Propbank comes of age—larger, smarter, and more diverse. In *Proceedings of the 11th Joint Conference on Lexical and Computational Semantics*, pages 278–288.

Vipula Rawte, Amit Sheth, and Amitava Das. 2023. A survey of hallucination in large foundation models.

Gözde Gül Şahin and Eşref Adalı. 2018. Annotation of semantic roles for the turkish proposition bank. *Language Resources and Evaluation*, 52:673–706.

Jaromir Savelka and Kevin D Ashley. 2023. The unreasonable effectiveness of large language models in zero-shot semantic annotation of legal texts. *Frontiers in Artificial Intelligence*, 6.

Karin Kipper Schuler. 2005. *VerbNet: A broad-coverage, comprehensive verb lexicon*. University of Pennsylvania.

Richard Shin and Benjamin Van Durme. 2022. Few-shot semantic parsing with language models trained on code. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5417–5425, Seattle, United States. Association for Computational Linguistics.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*.

Zhen Tan, Alimohammad Beigi, Song Wang, Ruocheng Guo, Amrita Bhattacharjee, Bohan Jiang, Mansooreh Karami, Jundong Li, Lu Cheng, and Huan Liu. 2024. Large language models for data annotation: A survey. *arXiv preprint arXiv:2402.13446*.

Ann Taylor, Mitchell Marcus, and Beatrice Santorini. 2003. The penn treebank: an overview. *Treebanks: Building and using parsed corpora*, pages 5–22.

Ashwini Vaidya, Martha Palmer, and Bhuvana Narasimhan. 2013. Semantic roles for nominal predicates: Building a lexical resource. In *Proceedings of the 9th Workshop on Multiword Expressions*, pages 126–131.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Transactions on Machine Learning Research*. Survey Certification.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, et al. 2023. Larger language models do in-context learning differently. *arXiv preprint arXiv:2303.03846*.

Leonie Weissweiler, Valentin Hofmann, Abdullatif Köksal, and Hinrich Schütze. 2022. The better your syntax, the better your semantics? probing pretrained language models for the english comparative correlative. *arXiv preprint arXiv:2210.13181*.

Peter West, Chandra Bhagavatula, Jack Hessel, Jena D Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. 2021. Symbolic knowledge distillation: from general language models to commonsense models. *arXiv preprint arXiv:2110.07178*.

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.

Nianwen Xue. 2006. A chinese semantic lexicon of senses and roles. *Language resources and evaluation*, 40:395–403.

Wajdi Zaghouani, Mona Diab, Aous Mansouri, Sameer Pradhan, and Martha Palmer. 2010. The revised arabic propbank. In *Proceedings of the fourth linguistic annotation workshop*, pages 222–226.

9. Language Resource References

A. Data Collection Details

For each verb, we leveraged COCA search to find instances of the verbs in transitive, intransitive, and middle voice usages. This allowed us to specify, for example, expected noun phrases in both the preverbal and postverbal positions for the transitive voice, the expected subject noun phrase and generally a prepositional phrase for intransitives, and finally the expected subject noun phrase and generally a postverbal adverbial phrase for the middle voice. From the search results, we attempted to select relatively simple sentences where the target verb was the matrix verb. We selected 2 instances of both transitive and intransitive usages, where one usage was relatively concrete (e.g., *...a fringe of activists broke some doors and windows of the halls and committed two minor assaults* and one usage was more abstract (e.g., *...this number broke all records for a single registration day.* Note that because PropBank senses are relatively coarse-grained, such usages are generally classed as the same sense as their semantics are similar as is the argument structure (Bonial et al., 2010).

Finding middle voice usages was more challenging, as these are less frequent and often isolated to advertising language. If we were unable to find the target verbs in middle voice usages in COCA, we completed secondary web searches and were able to find such usages in product reviews. Given that these usages are less frequent, we included and annotated only one middle voice usage for each verb, with the exception of the verb *parse*, for which we could find only one intransitive usage but many middle voice usages. Thus, we included one intransitive and two middle voice usages in addition to two transitive usages for it.

We acknowledge that the defining criteria of both intransitive and middle voice can be challenging. Although our defining criteria may be debatable, we note that we do not necessarily believe that a mis-classification would significantly alter the findings of our primary research question here, as we were primarily searching for distinct syntactic realizations of the same verb to determine if LLMs could track the semantic roles across those distinct realizations. Middle constructions are both particularly challenging and particularly interesting as they can be syntactically identical to intransitives (e.g., *This cake cuts beautifully*), but are semantically distinct as the *cake* is not doing the *cutting*.

B. Full Prompts

We present our full prompts, where curly brackets are placeholders for instances from our 35-sentence evaluation set.

Version 0 - “zero-shot-NoRoleset” (less info than given to an annotator)

Given the following verb and sentence, produce PropBank annotations of the verb sense and its arguments. Limit your annotation to the words in the sentence provided.

Annotate this:
Sentence:
Verb:

Version 1 - "3-examples-NoRoleset" (same info given to an annotator)
Given the following verb and sentence, produce PropBank annotations of the verb sense and its arguments. Limit your annotation to the words in the sentence provided.

Example 1:
Sentence: They went to India and Nepal, stayed in hostels and hiked mountains.
Verb: hike
Sense: hike.01 (walk for pleasure or exercise)
Arguments:
Arg0: They
Rel: hiked
Arg1: mountains

Example 2:
Sentence: Connor Kobal hikes regularly in Boulder Mountain Park.
Verb: hike
Sense: hike.01 (walk for pleasure or exercise)
Arguments:
Arg0: Connor Kobal
Rel: hikes
ArgM-TMP: regularly
ArgM-LOC: in Boulder Mountain Park.

Example 3
Sentence: This trail hikes through a portion of the historic area and then up to a ridge overlooking Stone Valley.
Verb: hike
Sense: hike.01 (walk for pleasure or exercise)
Arguments:
Arg1: This trail
Rel: hikes
ArgM-DIR: through a portion of the historic area and then up to a ridge overlooking Stone Valley.

Annotate this:
Sentence:
Verb:

Version 2 - 3-examples-Roleset (more info than given to an annotator)
Given the following verb and sentence, produce PropBank annotations of the verb sense and its arguments. Use the roleset information provided to produce the annotation. Limit your annotation to the words in the sentence provided.

Example 1:
Sentence: They went to India and Nepal, stayed in hostels and hiked mountains.
Verb: hike
Sense: hike.01 (walk for pleasure or exercise)
Roleset:
ARG0: causer of motion
ARG1: path of motion; location
Arguments:
Arg0: They
Rel: hiked
Arg1: mountains

Example 2:
Sentence: Connor Kobal hikes regularly in Boulder Mountain Park.
Verb: hike
Sense: hike.01 (walk for pleasure or exercise)
Roleset:
ARG0: causer of motion
ARG1: path of motion; location
Arguments:
Arg0: Connor Kobal
Rel: hikes
ArgM-TMP: regularly
ArgM-LOC: in Boulder Mountain Park.

Example 3
Sentence: This trail hikes through a portion of the historic area and then up to a ridge overlooking Stone Valley.
Verb: hike
Sense: hike.01 (walk for pleasure or exercise)
Roleset:
ARG0: causer of motion
ARG1: path of motion; location
Arguments:
Arg1: This trail
Rel: hikes
ArgM-DIR: through a portion of the historic area and then up to a ridge overlooking Stone Valley.

Annotate this:
Sentence:

Verb:
Sense:
Roleset:

Extending VerbNet's Verb-Specific Features to Enhance Selectional Preferences of Semantic Roles

Susan Windisch Brown
University of Colorado
Boulder, CO
susan.brown@colorado.edu

Abstract

This work proposes expanding the thematic role selectional preferences used in the lexical resource VerbNet as a way to increase the available semantic information in the resource, induce semantically-based subclasses for the more generic VerbNet classes, and create new links across classes. The addition of verb-specific features in the latest version of VerbNet provides a means for adding more specific selectional preferences based on the meaning of a class's individual member verbs. These features could refine both the instantiated class roles and the new implicit roles introduced in VerbNet version 4. We suggest 49 classes that would benefit from 111 verb-specific selectional preferences and explain how they would enhance VerbNet's semantic representations.

Keywords: semantic representations, VerbNet, thematic roles

1. Introduction

Deep learning has revolutionized natural language processing (NLP) in recent years, but problems with explanability and portability to low-resource languages or subject domains have led to the development of neurosymbolic methods. These new methods have made symbolic representations of meaning more relevant than ever for NLP. Lexical resources like VerbNet (Schuler, 2005), FrameNet (Baker et al., 1998) and PropBank (Kingsbury and Palmer, 2002) have a long history of contributing to NLP tasks that require rich semantic information, such as question answering, inferencing, and event and entity tracking. All three resources provide information on semantic roles, but VerbNet alone provides semantic representations for classes of verbs. These use Generative Lexicon subevent semantics (Pustejovsky, 1995, 2013) in a loosely neo-Davidsonian representation (Brown et al., 2019, 2022).

VerbNet's combination of syntactic and semantic regularities in the construction of its classes of verbs has resulted in some classes that are more syntactically than semantically coherent. Recent work (Kazeminejad et al., 2022) has added verb-specific features to the members of many VerbNet classes, allowing the formation of semantically coherent subclasses. We propose the addition of new verb-specific features that can both aid in that effort and enhance the semantic representations. These features would add more specific selectional preferences on the thematic roles based on the meaning of the individual member verbs (such as the Theme role in Build-26.1 class having the selectional preference FIBER for the verbs *knit* and *weave* but METAL for the verbs *hammer* and *forge*). These could refine both the traditional class roles and the implicit roles (e.g., V_Instrument) added to the semantic representations in VerbNet version 4. We suggest 49 specific classes that would benefit from 111 verb-specific selectional preferences and explain how they would enhance the semantic representations.

2. Background

VerbNet (Schuler, 2005; Schuler et al., 2009) is a large-scale English verb lexicon that uses similarities in verbs' syntactic and semantic behaviors to create hierarchical classes. Based on the classes created by Levin (1993), each class includes member verbs, general thematic roles that represent the arguments in the typical predicate-argument patterns of those verbs, and selectional restrictions on the class's thematic roles. The diathesis alternations that are the backbone of VerbNet's structure are listed in each class as syntactic patterns, and each syntactic pattern is accompanied by a semantic representation that incorporates the class's thematic roles (Bonial et al., 2011a,b).

The semantic representations list a series of semantic predicates, such as **has_location, desire** or **cause**, and an event variable **E**. The neo-Davidsonian representation uses the class's thematic roles as the arguments of the predicates and traces the progression of the event through subevent variables (Brown et al., 2022). The

Escape-51 class, for example, has a syntactic frame with the semantic representation seen in (1).

(1) *He came from France to Colorado.*

Agent V Initial_Location Destination

has_location(e_1, Theme, Initial_Location)
motion(e_2, Theme, ?Trajectory)[1]
¬**has_location**(e_2, Theme, Initial_location)
has_location(e_3, Theme, Destination)

The semantic representations are general enough to fit with all member verbs in a class. For classes with semantically very similar verbs, the representations can be quite specific. For other classes, the member verbs are semantically diverse, with only general semantic features applying to all verbs. For example, the Entity-Specific_COS (change of state)-45.5 class includes verbs as diverse as *blossom, spoil,* and *tarnish.* It has one thematic role (i.e., Patient), the selection preference +concrete on that role, and a simple, generic semantic representation that highlights the change in the Patient from not being in a particular state to being in that state:

(2) *The roses bloomed.*

¬**has_state**(e_1, Patient, V_Final_State)
has_state(e_2, Patient, V_Final_State)

This example illustrates the two types of thematic roles in VerbNet: those instantiated as arguments (e.g., Patient) and those that are incorporated into the meaning of the verb (e.g., V_Final_State). The first type are the roles that have been widely used for semantic role labeling (Shi and Mihalcea, 2005; Giuglea and Moschitti, 2006; Palmer et al., 2011), such as Agent, Patient, and Location. Each class lists the roles that get instantiated in sentences using the class's verbs. VerbNet has 39 roles, related hierarchically (Bonial et al., 2011b).

The other type of role was introduced with new semantic representations and is used to describe roles that are semantically necessary but that never appear as arguments in sentences using the class's verbs (Brown et al., 2022). They instead are incorporated into the verb itself, as indicated by the initial V_ in the role name. The V_Final_State role in the example above is a one example. Most of these uninstantiated roles are based on roles in the set of usual, instantiated roles. For example, the V_Instrument role in the Wipe_Instr-10.6.2 class (example verbs: *iron, shovel, sponge* corresponds to the instantiated thematic role Instrument in the Carve-21.2 class (example verbs: *dice, grind, slit*. V_Final_State is unusual in that there is no Final_State role in any VerbNet class. However, V_Final_State is used frequently as an argument in the semantic representation of change of state classes.

Although the syntactic and semantic generalizations provided by VerbNet classes have proved useful for numerous NLP tasks over the years, the option of accessing more specific semantic features for individual verbs or subsets of verbs in a class was often suggested as desirable (Gao et al., 2016; Clark et al., 2018). Kazeminejad et al. (2022) describes an effort to do that through the addition of fine-grained semantic features to individual verbs in a class. These features usually provide values for an attribute that several of a class's verbs share. For example, the Run-51.3.2 class has verbs (e.g., *scurry* and *whiz*) with the attribute VELOCITY and the value +FAST. For classes that are already semantically coherent but quite large, such as Run-51.3.2, these features can tie together the many verbs into helpful subgroups, such as all the verbs that refer to types of walking. For very general classes, such as Other_COS (change of state)-45.5, the features add more semantically coherent subgroups of verbs.

3. Adding Verb-Specific Selectional Preferences

VerbNet's regular 39 thematic roles are used across all its classes with the same, consistent definitions. Within each class, however, the thematic role may be further specified with a selectional restriction that indicates the type of entity that usually fulfills that role (Table 1). As explained in Palmer et al. (2016), the selectional restrictions are to be interpreted not as strict constraints but as preferences. Because VerbNet's roles are organized into a hierarchy in which more specific roles inherit all the qualities of their parent roles, the selectional preferences can be seen as a further subordinate level of that hierarchy.

Although the VerbNet selectional preferences have been used for various purposes in the past, such as disambiguating prepositional phrase attachment (Bailey et al., 2015) and metaphor detection (Wilks et al., 2013), some have found that they needed to use information from other resources to reach the desired level of specificity (Wilks et al., 2013; Di Fabio et al., 2019). For example, the creators of Verb Atlas (Di Fabio et al., 2019) used VerbNet thematic roles for their resource but substi-

[1] The question mark indicates a role that is semantically entailed and used in other syntactic frames within the class but not instantiated in this syntactic frame.

tuted WordNet hypernym synsets for the VerbNet selectional preferences on those roles to expand the possible set of preferences.

The current set of selectional preferences (Table 1) contain types that vary widely in the extent of their usage. The type ANIMATE is used with roles in 147 classes, ORGANIZATION in 127 classes, and CONCRETE in 75. However, 61% of types are used in 5 or fewer classes. The ubiquity of the very general selectional preferences (e.g., CONCRETE) results from the same semantic diversity of the verbs in some classes that lead to very generic semantic representations. In a class like Entity-Specific_COS-45.5, the most you can say about the types of entities that fulfill the Patient role (and still be true for every verb in the class) is that they are CONCRETE. For other classes, like Calibratible_COS-45.6.1, the Patient cannot be further constrained at all using the current set of selectional preferences.

selectional restriction	**No. of classes**	**selectional restriction**	**No. of classes**
abstract	4	int_control	25
animal	3	location	32
animate	147	machine	14
biotic	1	nonrigid	1
body_part	14	organization	127
comestible	6	plural	2
communication	10	pointy	1
concrete	75	reflexive	3
currency	5	region	20
elongated	2	solid	7
eventive	1	sound	1
force	1	substance	2
garment	1	vehicle	3
human	3	vehicle_part	1

Table 1: VerbNet selection restrictions

We propose adding selectional preferences to individual verbs within a class using the established verb-specific feature element. In the class Entity-Specific_COS-45.5, for example, a mix of existing selectional preferences (e.g., HUMAN and BODY-PART) and new ones (e.g., PLANT, METAL, and LIQUID) could be linked to individual verbs along with the role they restrict (see Table 2).

These additions would have several benefits:

- Increase the semantic information provided by VerbNet.
- Improve the semantic coherence of classes by creating subsets of verbs that share semantic features.
- Allow connections across classes for verbs in a particular semantic domain (e.g., verbs that pertain to food but that are housed in different classes, such as *bake* in the class Cooking-45.3, *eat* in the class Eat-39.1, and *spoonfeed* in the class Feeding-39.7, could be connected with a FOOD selectional preference for the Patient.
- Enhance the semantic representations when they are instantiated by particular verbs.

Class and Role	**Feature**	**Example verb**
Amuse; V_Emotion	positive feeling	cheer
	negative feeling	annoy
Calibratible_COS	+increase	rise
	+decrease	decline
	+fluctuate	swing
Remedy; Patient	**plant**	fertilize
	human	cremate
	animal	inseminate
	liquid	chlorinate
	air	humidify
Gobble; Patient	liquid	guzzle
	food	wolf

Table 2: Classes with existing verb-specific features that could act as selectional preferences (new proposed features in bold)

This final point was suggested in Brown et al. (2022). They suggested that the V_Direction role in the semantic representations for the Calibratible_COS-45.6.1 class could be refined by the verb-specific features when the representation is instantiated with items from text. For the sentence *The price of oil rose by 500% from $5 to $25.*, the arguments of the predicate **change_value** could be replaced with items from the text and with the verb-specific feature for *rise*, resulting in:

(3) **change_value**(e_2, INCREASE_V_DIRECTION, *500%*_Extent, *price*_Attribute, *oil*_Patient)

We suggest a slightly different format that uses a dot to combine the role and verb-specific feature, emphasizing the increased specificity of the role and the possibility of seeing it as a subtype of original role. Thus, the role in (3) would read V_DIRECTION.INCREASE. This format would also work well with the standard roles in VerbNet. When the specific verb is known, the representation can add the verb-specific feature to appropriate arguments in the representation. To apply this to one of the food-related verbs, the representation in the Gobble-39.3 class would change the generic Patient role to Patient.food when *gobble* is known to be the verb:

(4) *Cynthia gobbled the pizza.*
has_location(e_1, Patient.food, ?Source)
do(e_2, Agent)
body_process($\ddot{e}_3$, Agent)
motion($\ddot{e}_3$, Patient.food, ?Trajectory)
contain(e_4, Agent, Patient.food)
cause(e_2, e_3)

4. Method

We used a manual methodology to ensure highly reliable results. We started by considering classes that contain either of two VerbNet elements. One was existing verb-specific features, which often implicitly reference one of the thematic roles (e.g., the existing features INCREASE, DECREASE and FLUCTUATE in the Calibratible_COS-45.6.1 class. The only required task for those classes was to make that connection explicit. Most classes with role-related features, however, also seemed incomplete, such as Remedy-45.7, to which we suggest adding four additional features to restrict the Patient role (Table 2).

The other element that proved fruitful for identifying possible new features was the implicit role variation marked with V_. These roles by definition already point out that more specificity about the role could be found in the verb itself. Often a single attribute of the role was indentifiable in the verbs with a handful of values. For example, the Vehicle-51.4.1 class, which has such denominal verbs as *boat, bus,* and *jet*, uses a V_Vehicle role in its semantic representations. The verbs already have one of three features: MEDIUM_GROUND, MEDIUM_AIR, and MEDIUM_WATER. Additional features that refine the V_Vehicle role could be added, such as MOTOR VEHICLE, WATERCRAFT, and AIRCRAFT. These provide a middle level of specificity between V_Vehicle and the specific craft described by the verb itself, and they enable the creation of subsets of verbs based on vehicle type.

5. Proposed Features

We have identified 49 classes that could be enhanced with 111 selectional preferences as verb-specific features (see Appendix). The most common role that could be enriched with verb-specific selectional preferences is Theme, followed closely by Patient. Using the already verb-specific implicit roles that begin with V_ resulted in identifying several classes that would benefit from additional verb-specific features, such as Other_COS-45.45.4 and Remedy-45.7. Occasionally when one class was identified as eligible for new selectional preferences through its V_role (e.g., Sound_emission), it suggested a related class with no V_role (e.g., Substance_emission). A sample of classes and their proposed verb-specific selectional preferences are given in Table 3.

Class and Role	**Feature**	**Example verb**
Escape-51; Traject.	upward	rise
	downward	fall
	toward	approach
	away	recede
Calve-28.1; Patient	canine	pup
	feline	kitten
	bovine	calve
Create-26.4; Result	written_text	author
	music	compose
	dance	choreograph
	image	silkscreen
	artifact	fabricate
Preparing; V_fin._st.	cooked	bake
	fermented	brew
	mixed	mix
	burning	kindle

Table 3: Classes and verb-specific features that could act as selectional preferences

6. Future Work

We would like to validate our proposed features with a survey of English-language speakers, possibly on a crowd-sourced platform. Another possibility for validation or discovering new selectional preferences would be using Corpus Pattern Analysis (Hanks, 2013).

Semi-automating the process of discovering new selectional preferences would save time and money and could possibly be done by using LLMs. To test this idea, we queried Chat-GPT (3.5) on most of the verbs in the Entity-Specific_COS-45.5 class using the following query: "Can you group the following verbs according to the type of entities involved: flower, moult, rot, rust, germinate, oxidize, stagnate, sprout, wither, wilt, tarnish, swell, superate, tarnish, bud, atrophy, fester, crust, blossom, blister, spoil, erode, ebb? It created two groups, one with *germinate, sprout, wither, wilt, bud,* and *blossom* as verbs that involve plant life, and another with most of the other words as verbs that involve inanimate objects. Some verbs it ignored. These groupings are not perfect, but they do suggest some reasonable selectional preferences for the Patient role. GPT-4 would no doubt do a better job.

We would also like to test the utility of these features in a task like entity tracking. Kazeminejad et al. (2021) showed that VerbNet semantic representations improved performace on this task, suggesting that there might be further improvement with richer, verb-specific role preferences.

7. Conclusion

In this work, we have proposed the addition of verb-specific selectional preferences for certain VerbNet roles. The existing VerbNet element of verb-specific features on class member verbs provides a seamless way of incorporating this new information. We have argued that these new features would improve the semantic coherence of classes by creating subsets of verbs that share semantic features, allow connections across classes for verbs in a particular semantic domain, and enhance the semantic representations when they are instantiated by particular verbs.

8. Acknowledgements

We gratefully acknowledge the support of a sub award from RPI on DARPA KAIROS Program No. FA8750-19-2-1004. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA or the U.S. government. Thanks also to Ghazaleh Kazeminejad, Julia Bonn and Martha Palmer for their valuable input.

9. Bibliographical References

Daniel Bailey, Yuliya Lierler, and Benjamin Susman. 2015. Prepositional phrase attachment problem revisited: How verbnet can help. In *Proceedings of the 11th International Conference on Computational Semantics*, pages 12–22.

Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The berkeley framenet project. In *COLING 1998 Volume 1: The 17th International Conference on Computational Linguistics*.

Claire Bonial, Susan Windisch Brown, William Corvey, Martha Palmer, Volha Petukhova, and Harry Bunt. 2011a. An exploratory comparison of thematic roles in verbnet and lirics. In *Workshop on Interoperable Semantic Annotation*, page 39.

Claire Bonial, William Corvey, Martha Palmer, Volha V Petukhova, and Harry Bunt. 2011b. A hierarchical unification of lirics and verbnet semantic roles. In *2011 IEEE Fifth International Conference on Semantic Computing*, pages 483–489. IEEE.

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.

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*, 5:821697.

Susan Windisch Brown, Dmitriy Dligach, and Martha Palmer. 2014. Verbnet class assignment as a wsd task. In *Computing Meaning*, pages 203–216. Springer.

Peter Clark, Bhavana Dalvi, and Niket Tandon. 2018. What happened? leveraging verbnet to predict the effects of actions in procedural text. *arXiv preprint arXiv:1804.05435*.

Andrea Di Fabio, Simone Conia, and Roberto Navigli. 2019. Verbatlas: a novel large-scale verbal semantic resource and its application to semantic role labeling. 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 627–637.

Qiaozi Gao, Malcolm Doering, Shaohua Yang, and Joyce Chai. 2016. Physical causality of action verbs in grounded language understanding. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1814–1824, Berlin, Germany.

Ana-Maria Giuglea and Alessandro Moschitti. 2006. Semantic role labeling via framenet, verbnet and propbank. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 929–936. Association for Computational Linguistics.

Patrick Hanks. 2013. *Lexical analysis: Norms and exploitations*. Mit Press.

Ghazaleh Kazeminejad, Martha Palmer, Susan Brown, and James Pustejovsky. 2022. Componential analysis of english verbs. *Frontiers in Artificial Intelligence*, page submitted.

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.

Paul R Kingsbury and Martha Palmer. 2002. From treebank to propbank. In *LREC*, pages 1989–1993.

Beth Levin. 1993. *English verb classes and alternations: A preliminary investigation*. University of Chicago press.

Martha Palmer, Claire Bonial, and Jena D Hwang. 2016. 17 verbnet: Capturing english verb behavior, meaning, and usage. *The Oxford handbook of cognitive science*, page 315.

Martha Palmer, Daniel Gildea, and Nianwen Xue. 2011. *Semantic role labeling*. Morgan & Claypool Publishers.

J. Pustejovsky. 1995. *The Generative Lexicon*. Bradford Book. 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. ACL.

Karin Kipper Schuler. 2005. *VerbNet: A broad-coverage, comprehensive verb lexicon*. University of Pennsylvania.

Karin Kipper Schuler, Anna Korhonen, and Susan Brown. 2009. Verbnet overview, extensions, mappings and applications. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Tutorial Abstracts*, pages 13–14.

Lei Shi and Rada Mihalcea. 2005. Putting pieces together: Combining framenet, verbnet and wordnet for robust semantic parsing. In *International conference on intelligent text processing and computational linguistics*, pages 100–111. Springer.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Yorick Wilks, Adam Dalton, James Allen, and Lucian Galescu. 2013. Automatic metaphor detection using large-scale lexical resources and conventional metaphor extraction. In *Proceedings of the First Workshop on Metaphor in NLP*, pages 36–44.

10. Language Resource References

11. Appendix: Verb-Specific Selectional Preferences

Instrument.knife
Instrument.liquid
Instrument.noose
Instrument.poison
Material.fiber
Material.food
Material.metal
Material.wood
Patient.air
Patient.animal
Patient.animate
Patient.body_part
Patient.dance
Patient.eyebrows
Patient.eyelashes
Patient.feet
Patient.fingers
Patient.fire
Patient.food
Patient.forehead
Patient.hand
Patient.head
Patient.human
Patient.lips
Patient.liquid
Patient.metal
Patient.neck
Patient.plant
Patient.solid
Patient.teeth
Result.artifact
Result.image
Result.music
Result.written_text
Theme.aircraft
Theme.blood
Theme.body_part
Theme.decoration
Theme.excrement
Theme.fire
Theme.gas
Theme.image
Theme.label
Theme.liquid
Theme.motor_vehicle
Theme.numbers
Theme.pest
Theme.plant
Theme.plant_part
Theme.saliva
Theme.solid

Theme.surface_substance
Theme.sweat
Theme.urine
Theme.vocal_music
Theme.vomit
Theme.watercraft
Theme.words
Trajectory.away_from
Trajectory.downward
Trajectory.toward
Trajectory.upward
Destination.food
Destination.animal
Destination.clothing
Destination.furniture
V_Direction.decrease
V_Direction.fluctuate
V_Direction.increase
V_Emotion.negative_feeling
V_Emotion.positive_feeling
V_final_state.burning
V_final_state.cooked
V_final_state.fermented
V_final_state.in_pieces
V_final_state.mixed
V_final_state.pale_skin
V_final_state.straightened
V_final_state.unconscious
V_final_state.asleep
V_final_state.compressed
V_form.compressed
V_form.cut
V_form.elevation_gain
V_form.elevation_loss
V_form.pieces
V_form.surface_substance_removed
V_form.turn
V_Instrument.ears
V_Instrument.eyes
V_Instrument.nose
V_manner.bragging
V_manner.ceremonial
V_manner.complaining
V_manner.physical
V_manner.possibly_verbal
V_manner.verbal
V_Patient.bovine
V_Patient.canine
V_Patient.feline
V_sound.continuous
V_sound.punctual
V_sound.sharp
V_sound.soft
V_sound.vibrate
V_Theme.plant
V_Theme.seafood
V_vehicle.aircraft
V_vehicle.motor_vehicle
V_vehicle.sled
V_vehicle.watercraft

Chinese UMR annotation: Can LLMs help?

Haibo Sun, Nianwen Xue, Jin Zhao
Liulu Yue, Keer Xu, Yao Sun, Jiawei Wu
Brandeis University
{hsun, xuen, jinzhao, liuluyue, keerxu, yaosun, jiaweiwu}@brandeis.edu

Abstract

We explore using LLMs, GPT-4 specifically, to generate draft sentence-level Chinese Uniform Meaning Representations (UMRs) that human annotators can revise to speed up the UMR annotation process. In this study, we use few-shot learning and **Think-Aloud prompting** to guide GPT-4 to generate UMR sentence-level graphs. Our experimental results show that compared with annotating UMRs from scratch, using LLMs as a preprocessing step reduces the annotation time by two thirds on average. This indicates that there is great potential to integrate LLMs into the pipeline for complicated semantic annotation tasks.

Keywords: Uniform Meaning Representation, Large Language Models, Semantic Annotation

1. Introduction

Uniform Meaning Representation (UMR) (Gysel et al., 2021; Bonn et al., 2023) is a graph-based cross-lingual semantic representation that includes a sentence-level representation and a document-level representation. The sentence-level representation is based on Abstract Meaning Representation (AMR) (Banarescu et al., 2013) but has been extended to capture not only predicate-argument structures, word senses, and named entities as AMR does, but also aspectuality of events, person and number attributes of entities, and quantification. Its document-level annotation includes temporal and modal dependencies for events, as well as coreference relations for entities and relations. Such a comprehensive meaning representation is very demanding for human annotators in terms of the linguistic training they needed, as they have to internalize a large inventory of semantic concepts, relations, and attributes, and is very time-consuming to annotate.

One way to speed up the annotation process is to pre-parse the text into "draft" UMRs and have human annotators correct them. However, the parser needs a considerable amount of UMR-annotated data to train, and no large UMR training set exists yet. In UMR release 1.0 (Bonn et al., 2024), each language has fewer than a thousand sentences of annotated UMRs, and it is insufficient to train a parsing model with adequate performance. In this paper, we explore the use of Large Language Models (LLMs) to generate Chinese UMRs that human annotators can correct for the purpose of speeding up the annotation process. We investigated the question of whether using LLMs as a preprocessing step would reduce the amount of time required for human annotators to annotate the same amount of data compared to annotating UMRs from scatch. The answer to this question is determined by several factors. The most important factor is the quality of the UMRs generated by LLMs. If the UMRs generated by LLMs are of poor quality, the human annotator will need to spend so much time deconstructing the structure generated by the LLMs that they are better off starting from scratch. The second factor is the functionalities of the annotation tool used for UMR annotation. If the tool has functionalities that allow the copying of subgraphs of LLM-generated UMRs when constructing the correct UMR, this will lower the threshold of parsing accuracy needed for LLMs to have a positive impact. In our annotation experiments, we use UMR-Writer (Zhao et al., 2021; Ge et al., 2023), and this tool allows subgraphs to be copied and reused. Therefore, the primary factor will be the quality of the UMRs generated by LLMs.

Our experimental results show that using LLMs as a pre-processing step on average reduces the annotation time by about two thirds. The annotators reported that LLM-generated graphs often contain correct top-level structures and subgraphs that save annotator time annotating UMRs. An evaluation of LLM-generated parses shows that their qualities are slightly below that of initial human annotation, but not by far.

The rest of the paper is organized as follows. In Section 2, we describe Uniform Meaning Representation for Chinese to provide a concrete idea of how challenging it is to annotate Chinese UMRs. In Section 3, we introduce our approach to using LLMs to generate the draft graphs and detail several key challenges in constructing UMR graphs. In Section 4, we evaluate LLM-generated parses with respect to their well-formedness and overall evaluation scores against gold UMR graphs. We measure inter-annotator agreement (IAA) between human annotators and the time savings from annotating LLM-generated UMRs compared with UMR annotation from scratch, and we also summarize the feedback from human annotators that reveal the strengths and weaknesses of LLM-generated UMRs as the starting point for human annotation. Related work is discussed in Section 5 and we conclude in Section 6.

2. Chinese UMRs

In this section we briefly illustrate different aspects of UMR annotation with an example in (1). UMR is a representation for entire documents, not just individual sentences, so we show the UMR in Figure 1 for a text snippet of two sentences that forms a minimal document. Solid lines are labeled with semantic relations at the sentence level that include semantic roles and other semantic relations, as well as attributes, while the dotted lines represent relations at the document level.

(1) a. 新时代 集团于1995 年计划将 城市
New Era Inc. in 1995 plan BA City
电视 售与罗渣士 通讯
Television sell Rogers Communications
集团，以 集中 发展
Inc. , in order to focus on develop
新时代 电视 。
New Era Television .

“The New Era Inc. planned in 1995 to sell City Television to Rogers Communications Inc. in order to focus on the development of New Era Television.”

b. 罗渣士 当时 计划将 之从 有线
Rogers at that time plan BA it from cable
电视台 转型为 地面
TV station transform into terrestrial
广播 频道 ，并 加入十一 种
broadcast channel , and add eleven
语言 的 电视节目 。
language DE TV program .

“At that time, Rogers planned to transform it from a cable TV station into a terrestrial broadcast channel and add TV programs in eleven languages.”

Sentence-level representation The sentence-level representation includes word senses and predicate argument structures, named entity types, aspectual attributes of events, person and number attributes of entities. In Figure 1, 计划-01 in the first sentence represents the first sense of 计划 (“plan”), and it is a predicate that has two core arguments, *Arg0* which is a the company 新时代集团 (“New Era Group”), and *Arg1* 售-01, which has its own argument structure. It also has a non-core argument 发展-05 (“develop”) that serves as its purpose (*:purpose*, and a *date-entity* that serves as its temporal modifier (*:temporal*). In addition to arguments, since 计划-01 is an event, it also has an aspectual attribute that indicates it is a *State*. The semantic relations between the predicate and its arguments and attributes are represented as directed edges from the predicate to the argument or attribute.

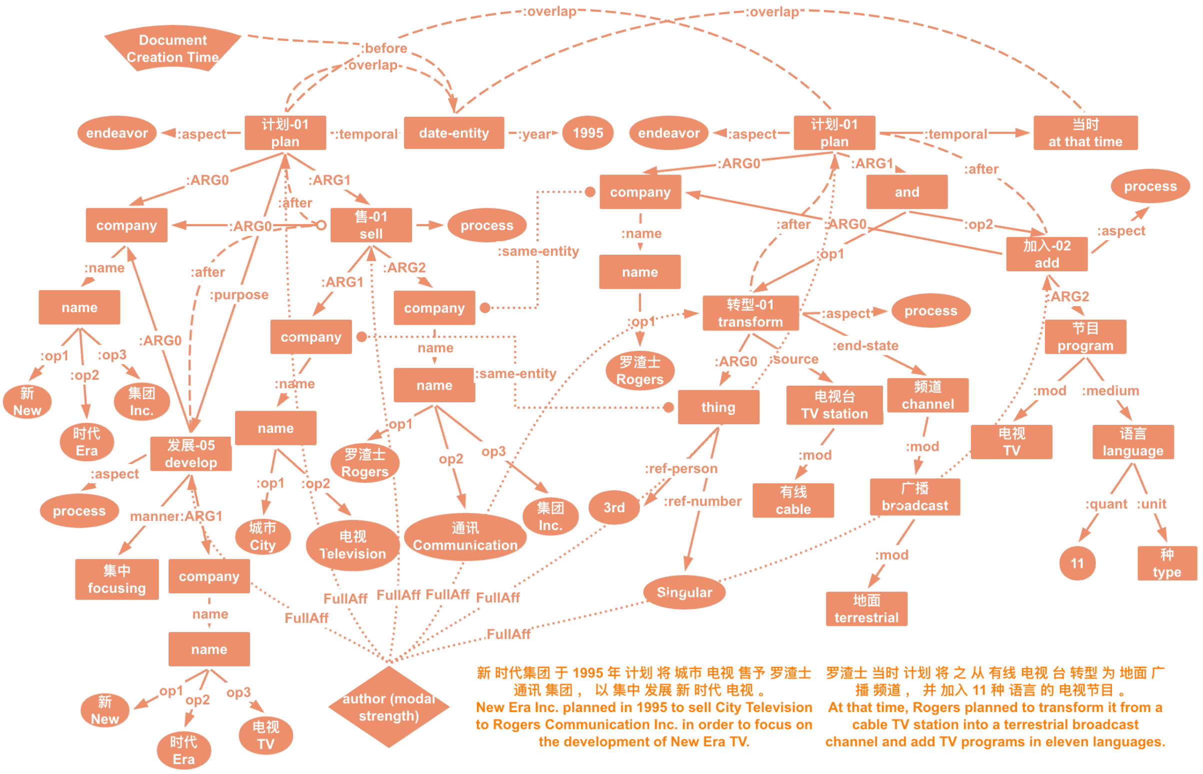
In addition to predicate-argument structures, UMR, following AMR, also represents named entity types. The named entity type is represented as a concept that has a list of strings that represent the actual name. For example, in Figure 1, 新~时代~集团~(“New Era Group”) is a name of the type *company*. Pronouns are typically represented as a concept with *person* and *number* attributes. For instance, 之 is a pronoun that maps to a *thing* concept with person attribute (*ref-person*) that has the value of *3rd*, and a number attribute (*ref-number*) that has the value of *Singular*.

Document-level representation Some semantic relations go beyond sentence boundaries, and these are represented as directed edges between a parent and a child, which can be (but not necessarily) in a different sentence. For example, the *thing* concept that derives from the pronoun 之 is coreferent with the *company* concept that refers to 罗渣士集团 (“Rogers Inc.”), and this is represented with the *same-entity* relation.

Temporal relations hold among events, between events and time expressions, and among time expressions. They are also represented as relations among concepts which can go beyond sentence boundaries or within the same sentence. As an example of temporal relations that go beyond sentence boundaries, the two instances of 计划-01 overlap with each other in terms of their temporal duration, just as the concepts 当时~in the second sentence overlap with the date-entity with the year 1995, as they refer to the same time period. As an example of temporal relations within the same sentence, 计划-01 (“plan”) is *before* 售-01 (“sell”) and 售-01 (“sell”) is *before* 发展-05 (“develop”).

Modal dependencies are relations between a *conceiver* or *source* and an event that indicate the level of certainty that the conceiver holds with respect to the event. In most cases the conceiver of an event is the author (AUTH), but it can also be other sources as well if the author cites a different source for the event.

It is very time-consuming for the annotator to annotate such a rich representation as UMR. We are interested in whether LLMs can be used to generate “draft” UMR graphs from raw text that annotators can correct to speed up the annota-



新 时代集团 于 1995 年 计划 将 城市 电视 售予 罗渣士 通讯 集团， 以 集中 发展 新 时代 电视 。
New Era Inc. planned in 1995 to sell City Television to Rogers Communication Inc. in order to focus on the development of New Era TV.

罗渣士 当时 计划 将 之 从 有线 电视 台 转型 为 地面 广播 频道， 并 加入 11 种 语言 的 电视节目 。
At that time, Rogers planned to transform it from a cable TV station into a terrestrial broadcast channel and add TV programs in eleven languages.

Figure 1: An example of a UMR graph for a mini-document of two sentences.

tion process. To make our study feasible, we conducted only experiments to generate sentence-level UMRs with LLMs.

3. Pre-parsing with LLMs

Prompt design is the key to the quality of LLM-generated UMRs. We explore three different prompting methods to observe their effect on the UMR parsing quality. We conduct our experiments in three settings: zero-shot, few-shot, and Think-Aloud. In the zero-shot setting, LLMs are not given any annotated examples, while in the few-shot setting, they are given a short document of 8 UMR-annotated examples. Finally, in the Think Aloud setting, in addition to the 8 examples, they are also given a step-by-step instruction of how the UMRs are annotated. We use GPT4 in all experiments.

3.1. Zero-shot setting

In the zero-shot setting, we give GPT4 the following prompt without any examples. Since UMR 1.0 was released after GPT-4 [1]was trained, we try to guide it to learn from AMR. However GPT-4 failed to generate any well-formed UMRs so we will not discuss it further.

> You are an expert linguistic annotator. You need to parse a given sentence into Uniform Meaning Representation, which is similar to Abstract Meaning Representation, but you need to name each variable starting with "s", followed by the number of sentence. All the tokens should only be from the sentence, and you must not hallucinate about any tokens or miss any tokens.

3.2. Few-shot setting

In the few-shot setting, we give GPT-4 the following instruction followed by UMRs of 8 sentences. When selecting the example UMRs, our aim is to have a good coverage of aspectuality attributes and modal strengths [2] that are new in UMR as they are absent in AMR, which has been around for longer periods of time and is therefore more accessible to LLMs. The instructions given to GPT-4 is as follows:

> You are a linguistic annotator. You need to follow the examples to parse a sentence into Uniform Meaning Representation step by step. You must name each variable starting with "s",

[1] The version we use is gpt-4-0125-preview.

[2] Modal strength is represented at the document level in UMR, but in most cases the conceiver or source is the author and can thus be annotated as a shorthand at the sentence level.

followed by the number of the sentence. All the tokens should only be from the sentence, and you must not hallucinate any tokens. You should identify the main verb as the head of the graph, and analyze the clauses recursively. You will also need to add “modal strength” to any predicates in the format “:modstr” with six possible values: [FullAff, PrtAff, NeutAff, FullNeg, PrtNeg, NeutNeg], and also add an aspect to any predicate in the form of “:aspect” with six possible values [Process, Endeavor, Performance, Activity, Habitual, State], and you will be shown how to use these values later. NEVER combine any tokens separated by space!

In the few-shot setting, no explanation is given to GPT-4, but we attempt to include examples of how common UMR concepts, attributes, and relations are represented. The following are UMR snippets that illustrate the representation of modal strength, aspectuality, and named entities and their relations.

Modal strength In the sentence 4, the main predicate is 讲 (“tell”). It is in a imperative mode, and under the modal verb 不能 (“cannot”), which makes its modal strength NeutNeg, meaning neutral negative.

(2) 这个关于 他 晋升 的 秘密 不能 给
this about he promote DE secret cannot to
任何人 讲 !
any person tell !

“You cannot tell anybody the secret that he got promoted!”[3]

```
(s1x / 讲-01[“tell”]
     :mode imperative
     :modstr NeutNeg
     ...)
```

Aspectuality An example of aspectuality represented in the UMR is also provided in 3:

(3) ... 临近 演唱会 尾声
... approaching concert end

“... near the end of the concert”

```
... (s2x2 / 临近-01[“approaching”]
          :ARG0 (s2x3 / 演唱会 [“concert”])
          :ARG1 (s2x4 / 尾声 [“end”])
          :aspect State
          :modstr FullAff
          ...)
```

Named entities in appositive constructions We also provided GPT-4 some common patterns in UMR annotation, such as appositive constructions that involve a named entity of type *individual-person* that has a particular type of position in some organization, which is often also a named entity:

(4) 美国前 总统 克林顿
US former president Clinton

```
(s41i2 / individual-person
     :name(s41n / name
           :op1 ” 克林顿”[“Clinton”])
     :ARG1-of (s41h / have-org-role-91
         :ARG2 (s41c / country
                 :name (s41n2 / name
                     :op1 ” 美国”[“US”]))
         :ARG3(s41x3 / 总统 [“president”]
                 :mod (s41x4 / 前 [“former”]))))
```

The UMR inherits some of the named entity types from AMR but also adds quite a few new ones that reflect the different types of named entity in different cultures. This type of structure is very common in the data and if they can be correctly generated by LLMs, it would be a big help for human annotators who post-edit these UMRs.

3.3. Think-Aloud Prompting

Inspired by the research on using Chain-of-thought (Wei et al., 2022) when prompting LLMs to perform natural language tasks, we experimented with asking GPT-4 to follow the Think-Aloud Protocol (Jääskeläinen, 2010) to instruct it to explicitly output the thought process before outputting the final answer. This technique is originally an experiment protocol in psychology and ergonomics with human subjects. It asked human subjects to “think-aloud” - voicing their thoughts explicitly when or after completing a given task - to make researchers aware of how and why they take particular actions (Oh and Wildemuth, 2009; Wolcott and Lobczowski, 2021). Because the reasoning process is expressed in natural language and human annotation is guided by latent thoughts, we hypothesize that the reasoning process can serve as an intermediary from texts to semantic graphs. We designed the though process to particularly address the following issues.

Mismatch between the linear order of the sentence and the hierarchical structure of UMR graphs While a sentence is presented linearly from left to right, its UMR graph is a hierarchical structure with no meaning attached to the linear order. This mismatch may worsen the tendency of LLMs to hallucinate or ignore certain parts of

[3] The glossing abbreviations used in this paper are: DE: possessive or genitive marker

the input sentence. We therefore instruct GPT-4 to “think” hierarchically by selecting the main predicate first and then attending to its arguments. If the argument itself is another predicate, then GPT-4 is asked to recursively identify its arguments in a depth-first manner similar to the annotating process of human annotators. This is illustrated in (5).

(5) **Prompt**:Parse this sentence: 该周报也曾经短暂的拥有过一个属自己的电视节目。
Prompt translation: The weekly report also once briefly had its own television program.
Exemple response: 这是第 3 句话。
我们需要先判断这句话的句式，然后分析这句话中每个词语之间的依存句法关系。
该句子是一个陈述句，top node 是“拥有”
“拥有”的主语是“周报”，“周报”的修饰是“该”，因此“拥有”:arg0“周报”(s3x2)
“拥有”的宾语是“节目”，而“节目”的定语是“电视”，而这里“电视”是“节目”的媒介，所以“媒介”:medium“电视”
“节目”属于“自己”，而“自己”在这里是一个pronoun，指代的是“周报”，因此“属”:arg1 s3x2
… …
Response translation: This is the third sentence. We need to first determine the sentence structure of this sentence and then analyze the dependency syntax relationships between each word in the sentence.
The sentence is a declarative sentence, with ’owning’ as the top node.
The subject of ’owning’ is ’weekly report,’ and ’the’ modifies ’weekly report,’ hence ’owning’: arg0 ’weekly report’ (s3x2).
The object of ’owning’ is ’program,’ and the attribute of ’program’ is ’television,’ where ’television’ serves as the medium of ’program,’ so ’medium’: medium ’television.’
’Program’ belongs to ’oneself,’ and ’oneself’ here is a pronoun, referring to ’weekly report,’ thus ’belong’: arg1 s3x2
...

Identifying the abstract concepts for discourse relations In UMR, discourse relations are represented as a close set of abstract concepts that form a lattice so that different languages can select discourse relations of different levels of granularity for that language. Discourse relations are represented as abstract concepts because they are often implicit and can only be inferred from the larger context of the discourse. As such it is easy for LLMs to miss them. We attempt to guide LLMs to detect them with Think-Aloud prompting. This is illustrated in (6).

(6) **Prompt**: 目前《世界新闻周报》的纸本已经停刊，不过仍继续发行网络版。
Translation: Currently, the print edition of “World News Weekly” has been discontinued, but the online version is still being published.
Exemple response: 该句子是一个转折句，因此top node 是 but-91
前半句的谓语是“停刊”，因此 but-91 :arg1 停刊
… …
后半句的谓语是“继续”，因此 but-91 :arg2 继续
Response translation: The sentence is a contrastive sentence, therefore the top node is but-91. The predicate of the first half of the sentence is ”discontinued”, thus but-91 :arg1 is ”discontinued”. The predicate of the second half of the sentence is ”continues”, thus but-91 :arg2 is ”continues”.

Identifying named entities and their types. The named entity types of UMR also form a hierarchy, and they extend the AMR named entity types to cover a wide range of named entities from different cultures of the world. Some of them may have never been seen by GPT-4, and we instruct it to pay particular attention to them, as illustrated in (7).

(7) **Prompt**: 《世界新闻周报》是一份已经停止发行的美国超市小报。
Translation: The World News Weekly is an American supermarket tabloid that has ceased publication.
Example response: 判断句的主语，是“《世界新闻周报》”，是【命名实体】，是一种 newspaper，因此是 named entity，类型是 newspaper，首字母是 n，因此变量名是 s1n，因此 s1i2 :ARG1 newspaper :name 《世界新闻周报》。
Response translation: The subject of the judgment sentence is ’The World News Weekly’, which is a named entity, a type of newspaper. Therefore, it is a named entity, with the type being newspaper, and its first letter is ’n’, thus the variable name is ’s1n’. Therefore, ’s1i2 :ARG1 newspaper :name The World News Weekly’.

Identifying predicates and annotating their aspects and modal strength As aspect and modal strength are only relevant to predicates, to annotate the aspect and model strength values correctly, the first step is to identify the predicates. Aspect annotation is difficult for human annotations due to the lack of explicit aspect markers for most predicate instances, it is even difficult for human annotators. The modal strength value also has different manifestations in the Chinese language, and they can be derived from modal verbs, certain adverbs, or quoted speech. So we designed instructions to guide GPT-4 to pay attention to the right places, as illustrated in (8).

(8) **Prompt**:……阿扎扎称：……，结果竟在拿破仑头骨中发现了一枚无法解释的神秘芯片。
Prompt translation:...Azaza said: ..., surprisingly, a mysterious chip that cannot be explained was found in Napoleon’s skull.
Exemple response: 【“解释”是一个谓词，它的语法体标记 (:aspect) 只能从 state, performance, activity, habitual, endeavor, process 中选择，它

的语气强度 (modality strength)(:modstr) 只能从 FullAff, PrtAff, NeutAff, FullNeg, PrtNeg, NeutNeg 中选择】。“解释”是一个动作但不一定有结果和开始，因此“解释”:aspect Process；由于解释有一个“无法”作为修饰，表达的是否定意义，而“解释”来自于说话人的内容，无法确定其真实性，只能作推断，因此“解释”:modstr PrtNeg；同时，由于“解释”来自于说话人的内容，需要引用到上一个谓词，因此“解释”:QUOT “称”
Response translation: The verb ’explain’ has its grammatical aspect marker (:aspect) that can only be chosen from state, performance, activity, habitual, endeavor, process, and its modality strength (:modstr) can only be chosen from FullAff, PrtAff, NeutAff, FullNeg, PrtNeg, NeutNeg. ’Explain’ is an action that may not necessarily have a result or even start, therefore ’explain’: aspect Process; since ’explain’ is modified by ’unable to’, expressing a negative meaning, and ’explain’ comes from the speaker’s content, its truth cannot be determined, only inferred, therefore ’explain’ :modstr PrtNeg; meanwhile, since ’explain’ comes from the speaker’s content, it needs to refer to the previous predicate, therefore ’explain’ :QUOT ’said’.

4. Experiments

We conducted experiments to answer three questions: (i) How does GPT-4 perform in generating UMRs in a few shot and Think-Aloud settings? (ii) How is GPT-4 faring in comparison with human annotators? (iii) Does it take less time for human annotators to correct GPT-generated UMRs than annotating from scratch? We answer these questions through quantitative evaluations and also through qualitative analysis.

4.1. Experiment setup

We selected two articles published in the latter half of 2023 to conduct experiments on to make sure these articles were not included as part of the training data for GPT-4. The articles were chosen from authoritative news agencies to guarantee its grammaticality and factuality. Both articles have 26 sentences so that they can be finished in a reasonable amount of time.

The human annotation experiments are performed by four annotators. These annotators do not have extensive linguistic backgrounds but have taken linguistic courses. In order to have fair comparison of annotation speed under the two conditions, annotating from scratch vs annotating from GPT-generated UMRs, we need to make sure that the same annotator does not annotate the same article twice. We divide the four annotators into two groups, with two annotators in each group. We first have each group annotate one of the two articles from scratch, and then switch to annotate the other article from GPT-generated UMRs. After they finished annotating the articles from scratch, each group met to discuss their differences and arrived at a consensus annotation that we designate as the gold annotation.

4.2. Quality of GPT-generated UMRs

We used GPT-4 to generate UMRs for the two articles in few-shot and Think-Aloud settings, each with temperatures of 0 and 0.7. We thus have four UMR graphs generated under four conditions: few-shot at temperatures of 0 (0F) and 0.7 (7F) , Think-Aloud at 0 (0T) and 0.7 (7T).

GPT-4 generated fully well-formed UMRs under condition 0T, but there are occasional format errors under other conditions, and the higher temperature (0.7) leads to many more format errors. These include:

1. Quoted reentrancy, such as *:ARG0 (s24x)* where the variable should not be bracketed;
2. Duplicated variable names;
3. Extra right brackets ;
4. Unclosed brackets;
5. Multiple unconnected graphs in one sentence;
6. Unrelated content, extra explanations after the graph;

The four GPT-generated UMRs, after corrections of format errors, are tested against the gold data with four AnCast metrics (Sun and Xue, 2024): Labeled Relation F1 (LRM), Unlabeled Relation F1 (ULRM), Weighted Relation F1 (WLRM), and Concept F1 (CM), as well as Smatch (Cai and Knight, 2013) and Smatch++ (Opitz, 2023). All the scores are macro-averaged among the 26 sentences in an article, and the results for each article are presented in Table 1.

As can be observed from Table 1, the SMatch (SM) scores for the two articles are in the 40 and 50 percentage range, while the LRM scores, a harsher metric as a relation matches only if the concepts in the relation match as well, are in the 30 and 40 percentage range. There is no clear pattern as to which of the four conditions fares better, but some conditions work better for some sentences while other conditions work better for others.

4.3. Performance of human annotators

Each article is annotated by two pairs of annotators, with the first pair annotating from scratch and the second pair annotating from GPT-generated UMRs. The draft UMRs used for our human annotation experiment is generated with Think-Aloud prompting at temperature 0, and are fully well

Article 1	A-A	A1-G	A2-G	0F-G	7F-G	0T-G	7T-G
CM	78.52	93.72	88.32	79.03	75.93	65.61	81.90
ULRM	53.97	78.92	70.08	46.93	42.28	48.20	47.00
WLRM	53.05	77.64	70.66	41.16	37.62	42.72	42.88
LRM	52.00	78.08	68.66	43.61	38.47	44.44	43.00
SM	60.85	80.08	75.38	55.58	52.69	54.73	53.92
SM++	60.45	79.93	75.06	55.12	52.18	53.99	53.51
Article 2	A-A	A3-G	A4-G	0F-G	7F-G	0T-G	7T-G
CM	61.65	97.06	77.86	65.82	64.35	72.02	72.51
ULRM	42.88	85.31	42.44	34.35	34.96	31.37	33.69
WLRM	45.17	90.12	43.84	30.46	32.45	32.39	33.28
LRM	40.77	84.97	40.43	31.23	32.12	28.72	31.30
SM	53.15	87.96	55.00	46.81	46.85	41.62	44.35
SM++	53.33	88.23	54.70	47.15	46.74	41.26	44.12

Table 1: Inter-Annotator Agreement (IAA) and Automatic UMR Parsing Accuracy. The scores in the upper half are for Article 1, and that in lower half are for Article 2. The scores in the left half are for the IAA between human annotators and the scores of each annotator pair against the gold graph while the scores in the right half are for the GPT-generated UMRs against gold graphs. The leftmost column indicates the evaluation metrics we used: CM (concept match) measures the F1-score of the set of concepts annotated in two graphs; ULRM (unlabeled relation match) measures the F1-score of parent-child concept pairs in two graphs; LRM (labeled relation match) takes the relation labels into account when measuring the F1 of the parent-child concept pairs; WLRM (weighted labeled relation match) is a weighted version of LRM with more weight given to nodes that have more descendants. The top row indicates what is measured: A-A means inter annotator agreement; A1/3-G and A2/4-G compares the UMRs by two annotators in each article with gold graph; 0F-G, 7F-G, 0T-G, 7T-G: the four LLM parses under different setting compared to the gold graphs. The definitions of settings are explained in section 4.2. The gold graph is obtained by merging the two annotations after a discussion between the two annotators. The discrepancy in scores between the gold graphs and those of different annotators reflect the varying levels of proficiency in UMR annotation for the annotators. Article 2 is more colloquially written than 1, which adds to the difficulty of annotation and results in a lower IAA.

formed. The IAA is calculated based on the annotations from scratch. From Table 1, we can see that the IAA is 60.85 % and 53.15% respectively for the two articles in terms of the SMatch score, and 52% and 40.77% in terms of LRM. Since the annotators are still under training, the IAAs are acceptable. We also computed the average accuracy for each pair of annotators by comparing their annotations with the gold graphs, and as can be seen from the table, the scores for all metrics tend to be higher than the IAA, which is not surprising since the gold graph is the consensus graph that is closer in similarity to each of the annotations.

From Table 1, we can also see that the accuracy of GPT-generated graphs are not substantially lower than the IAA of human annotators. In particular, GPT-generated UMRs are particularly strong in terms of Concept F1, while human annotators are better at judging relations in UMR, as reflected in the much higher scores in terms of LRM and ULRM.

Our users used UMR Writer (Zhao et al., 2021) to annotate the sentence level UMRs from scratch. UMR Writer provides annotators with segmented sentences and dropdown menus for relation labels, abstract concepts, aspect attributes, modal strength values, and other items in the UMR vocabulary. When annotating from scratch, the users need to manually select the segmented words, and then choose the corresponding item in the UMR vocabulary from the dropdown menus to assemble the UMR graph piece by piece; if there is already annotated content, the annotator can use the “move” function to rearrange the subgraphs.

Revising GPT-generated UMRs vs annotating from scratch To answer the question of whether annotating from GPT-generated UMRs can speed up the annotation process, we asked the annotators to carefully record their time when annotating from scratch and from draft graphs, and the results are shown in Table 2. The result shows that annotators on average spend only 1/3 of the time when annotating from draft UMR graphs compared with annotating from scratch. This indicates a significant improvement in efficiency when LLMs are incorporated into the UMR annotation pipeline as a preprocessing step.

Article	Annotator	From Scratch	Annotator	From Draft Graphs	Ratio
1	A1	8h57min	A3	2h47min	3.19
	A2	9h03min	A4	2h52min	
2	A3	6h49min	A1	2h51min	2.61
	A4	8h47min	A2	3h08min	

Table 2: A comparison between the times needed for annotation from scratch and from draft graphs. The method for calculating the ratio involves computing the average annotation time for each sentence, and then taking the average between the two annotators.

After the annotation, we asked the annotators for feedback on what contributed to the speedup in annotation when GPT-generated UMRs are used as the starting point for manual correction and on what the main issues GPT-generated UMRs still have in order to inspect the acceleration with finer granularity. The main advantages of annotating from GPT-generated UMRs are that (i) especially for simple and short sentences, the GPT-generated UMRs are very accurate and are able to correctly annotate many concepts, abstract and concrete, as well as attributes, (ii) Many subgraphs that correspond to common patterns are correctly annotated, (iii) Reentrancies are correctly identified for the most part, and (iv) Some GPT-generated UMRs suggest interpretations of the sentence that even human annotators find difficult.

The annotators also identify areas where GPT-4 typically makes mistakes. They point out that GPT-4 often makes mistakes for long and complicated sentences that involve mulitple clauses, and often messes up the discourse relations between the clauses. GPT-4 also often fails to properly decompose long compounds words, which are very common in Chinese, into concepts. Finally, GPT-4 still tends to hallucinate relation labels that are not in UMR. This means that annotators would have to correct these mistakes when annotating from GPT-generated UMRs.

5. Related Work

Preprocessing in annotation is not a new idea, and it has been deployed in annotation tasks before. Especially for complicated annotation tasks, it has been shown to speed up annotation in treebanking (Chiou et al., 2001). Prior to the availability of LLMs, in order for pre-processing tool to produce annotation of high enough quality, it has to be trained on a significant amount of human annotated data. That means that before such a machine preprocessing - human correction process can start, a significant amount of data, sufficient to train a reasonably accurate machine learning model, has to be annotated by human annotators from scratch first. The availability of LLMs makes it possible to start this process much earlier if they can be prompted to generate the annotation without already having a significant amount of annotated data.

There is also prior work on using LLMs to generate Abstract Meaning Representations (AMRs) using GPT-4 (Ettinger et al., 2023) and comparing the quality of AMRs generated by LLMs with AMR parsers trained on million-plus human annotated AMRs. Their results show that while LLMs have shown some capability of generating AMRs, the quality of AMRs they generated are still substantially below that of state-of-the-art AMR parsers trained on large quantities of human annotated AMRs. They did not conduct experiments on whether the AMRs LLMs generated can help reduce the annotation time compared with human annotation from scratch.

6. Conclusion and Future work

In this paper, we investigated the question of whether LLMs, specifically GPT-4, can be used to speed up UMR annotation. Although the data set we used is relatively small, with only two articles, it is safe to conclude that incorporating LLMs into the annotation pipeline as a preprocess step can significantly reduce the amount of time (and cost) in UMR annotation. We also found that the accuracy of GPT-generated UMRs is not very far from the IAA from human annotators, with the caveat that the human annotators are still undergoing the training phase. The experiment on which prompting strategy produces the most accurate UMRs is inconclusive and additional experiments are needed to get a definitive answer. Future work also includes deploying LLMs to get modality, temporal dependency and coreference annotation at the document for UMR annotation.

Acknowledgements

This work is supported by grants from the CNS Division of National Science Foundation (Awards no: NSF_2213804) entitled “Building a Broad Infrastructure for Uniform Meaning Representations”. Any opinions, findings, conclusions or recommendations expressed in this material do not necessarily reflect the views of NSF. We also wish to extend our appreciation to Cloudbank, which provided an indispensable computational resource for our experiments.

Limitations

The data set used in our experiments are relatively small, with only two documents that each have less than 30 sentences. However, we are confident with our conclusion that using LLMs as a preprocessing step speeds up UMR annotation.

7. Bibliographical References

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 *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria. Association for Computational Linguistics.

Julia Bonn, Matthew Buchholz, Jayeol Chun, Andrew Cowell, William Croft, Lukas Denk, Sijia Ge, Jens E. L. Van Gysel, Jan Hajič, Kenneth Lai, James H. Martin, Skatje Myers, Alexis Palmer, Martha Palmer, Benet Post, James Pustejovsky, Kristine Stenzel, Haibo Sun, Zdeňka Urešová, Rosa Vallejos Yopán, Nianwen Xue, and Jin Zhao. 2024. Building an infrastructure for uniform meaning representations. In *Proceedings of LREC-COLING 2024*.

Julia Bonn, Andrew Cowell, Jan Hajic, Alexis Palmer, Martha Palmer, James Pustejovsky, Haibo Sun, Zdenka Uresova, Shira Wein, Nianwen Xue, et al. 2023. UMR annotation of multiword expressions. In *Proceedings of the 4th International Workshop on Designing Meaning Representations*.

Shu Cai and Kevin Knight. 2013. 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.

Fu-Dong Chiou, David Chiang, and Martha Palmer. 2001. Facilitating treebank annotation using a statistical parser. In *Proceedings of the first international conference on human language technology research*.

Allyson Ettinger, Jena Hwang, Valentina Pyatkin, Chandra Bhagavatula, and Yejin Choi. 2023. "you are an expert linguistic annotator" : Limits of llms as analyzers of abstract meaning representation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8250–8263.

Sijia Ge, Jin Zhao, Kristin Wright-Bettner, Skatje Myers, Nianwen Xue, and Martha Palmer. 2023. UMR-Writer 2.0: Incorporating a new keyboard interface and workflow into UMR-Writer. In *Proceedings of the 17th Linguistic Annotation Workshop (LAW-XVII)*, pages 211–219.

Jens E. L. Van Gysel, Meagan Vigus, Jayeol Chun, Kenneth Lai, Sarah Moeller, Jiarui Yao, Timothy J. O'Gorman, Andrew Cowell, William Croft, Chu Ren Huang, Jan Hajic, James H. Martin, Stephan Oepen, Martha Palmer, James Pustejovsky, Rosa Vallejos, and Nianwen Xue. 2021. Designing a Uniform Meaning Representation for Natural Language Processing. *Künstliche Intelligenz*, pages 1–18.

Riitta Jääskeläinen. 2010. Think-aloud protocol. *Handbook of translation studies*, 1:371–374.

Sanghee Oh and B Wildemuth. 2009. Think-aloud protocols. *Applications of social research methods to questions in information and library science*, pages 178–188.

Juri Opitz. 2023. SMATCH++: Standardized and extended evaluation of semantic graphs. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1595–1607, Dubrovnik, Croatia. Association for Computational Linguistics.

Haibo Sun and Nianwen Xue. 2024. Anchor and broadcast: An efficient concept alignment approach for evaluation of semantic graphs. In *Proceedings of the 30th International Conference on Computational Linguistics*. To appear.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Michael D Wolcott and Nikki G Lobczowski. 2021. Using cognitive interviews and think-aloud protocols to understand thought processes. *Currents in Pharmacy Teaching and Learning*, 13(2):181–188.

Jin Zhao, Nianwen Xue, Jens Van Gysel, and Jinho D Choi. 2021. UMR-Writer: A web application for annotating uniform meaning representations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 160–167.

Accelerating UMR Adoption: Neuro-Symbolic Conversion from AMR-to-UMR with Low Supervision

Claire Benét Post[*], Marie McGregor[*], Maria Leonor Pacheco, Alexis Palmer
University of Colorado Boulder
{benet.post, marie.mcgregor, maria.pacheco, alexis.palmer}@colorado.edu

Abstract

Despite Uniform Meaning Representation's (UMR) potential for cross-lingual semantics, limited annotated data has hindered its adoption. There are large datasets of English AMRs (Abstract Meaning Representations), but the process of converting AMR graphs to UMR graphs is non-trivial. In this paper we address a complex piece of that conversion process, namely cases where one AMR role can be mapped to multiple UMR roles through a non-deterministic process. We propose a neuro-symbolic method for role conversion, integrating animacy parsing and logic rules to guide a neural network, thus minimizing human intervention. On test data, the model achieves promising accuracy, highlighting its potential to accelerate AMR-to-UMR conversion. Future work includes expanding animacy parsing, incorporating human feedback, and applying the method to broader aspects of conversion. This research demonstrates the benefits of combining symbolic and neural approaches for complex semantic tasks.

Keywords: Uniform Meaning Representation, Abstract Meaning Representations, Animacy Parsing, Neuro-Symbolic Learning, Low-Resource Setting

1. Introduction

Meaning representation graphs are hierarchically structured discrete representations of meaning that allow for sentence and document-level meanings to be abstracted away from syntactic structures. They utilize graphical representations where sentences with similar meanings share similar graph structures, even if worded differently. Abstract Meaning Representation (AMR) graphs model sentence-level meanings (Banarescu et al., 2013), and although they can be applied to different languages, the annotation guidelines are closely tied to English, for instance, by not supporting polysynthetic languages. Uniform Meaning Representation (UMR) (Gysel et al., 2021) addresses this limitation by extending AMR to support both sentence and document-level representations, and providing a typologically-motivated, language-agnostic schema for representing meaning.

Direct human annotation of texts with UMR graphs is time-consuming and requires considerable domain expertise. In order to speed up production of data, we take a first step towards automatically converting existing AMR annotations[1] to the more detailed, richer UMR schema.[2] Figure 1 shows a side-by-side comparison. Generating a preliminary graph for annotators to refine, even if noisy, could significantly reduce the human effort required. There are roughly 60,000 annotated English AMR sentences, and parallel UMR annotations previously existed for only about 200 of those. This means we have minimal parallel data from which to train a model on the conversion task.

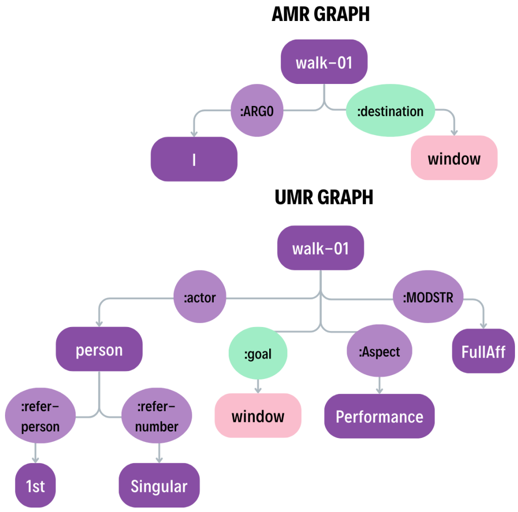


Figure 1: Example of one type of graph conversion of the AMR :destination role to the UMR role :goal in "I walked up to the window".

This paper presents an automated method for partial graph conversion, specifically addressing non-deterministic changes arising from AMR to UMR. [3]. AMR graphs contain individual semantic rolesets that convert into multiple rolesets in UMR. These rolesets between AMR and UMR are known as "split-roles" and contain a non-deterministic,

[*] Equal Contribution

[1] AMR site: https://amr.isi.edu/.

[2] UMR site: https://umr4nlp.github.io/web/

[3] Our codebase can be foud at: https://github.com/clairepost/AMRtoUMR

5th Workshop on Designing Meaning Representations (DMR 2024) @LREC-COLING-2024, pages 140–150
21 May, 2024.

1:many relationship. This non-determinism motivates our focus on these rolesets, as previous work suggested human annotation would be necessary for their conversion (Bonn et al., 2023).

To address this challenge, we propose a modular, neuro-symbolic framework that utilizes an animacy parser to assist logic rules in automatically determining split roles, minimizing the need for human input in UMR annotation. Our framework combines the flexibility afforded by neural methods to identify patterns in raw data, with a way to promote the schematic constraints of the conversion task. To train and evaluate our framework, we curate a dataset of 587 manually annotated role conversions and 10,635 weakly annotated role conversions, spanning 14 different split role types.

This paper focuses on English AMRs, but the methods presented can be adapted to AMRs in other languages. This adaptability stems from the inherent language-agnostic nature of the underlying graph structure. However, future work in other languages may encounter additional challenges, particularly in accessing an animacy parser. While adapting the approach for AMRs in languages like Chinese may be more feasible due to the availability of resources, languages with limited NLP resources, such as Cherokee, may pose greater difficulties. We limit the scope of these AMR-to-UMR conversions to sentence-level, leaving document-level graph creation for future work.

In summary, we make the following contributions: (1) We frame AMR to UMR conversion as a prediction task, (2) We curate and annotate a dataset focused on split role conversion from AMR to UMR, (3) We propose an extensible, modular framework that combines neural networks and domain knowledge in the form of rules to make this prediction, and (4) We show that we can accurately predict the majority of the non-deterministic roles with limited supervision.

2. Related Work

While AMR has established itself as a powerful tool for semantic representation, its limitations in handling low-resource languages and complex linguistic phenomena hinder its broader applicability. These limitations include challenges with morphology, like polysynthesis, and capturing relationships beyond the sentence level in document-level annotations. UMR, recently proposed by Gysel et al. (2021), offers a compelling alternative with a richer semantic framework and multilingual focus. It introduces document-level representations alongside sentence-level analysis, capturing more nuanced semantic information such as co-reference, temporal, and modal dependencies that go beyond sentence boundaries. However, despite its advantages, UMR adoption is currently hampered by the scarcity of annotated data. This section positions our work within the context of related efforts bridging the gap between AMR and UMR, particularly through automated conversion approaches. Additionally, our efforts complement the work on bootstrapping UMR annotations for low-resource languages, as presented in (Buchholz et al., 2024). This paper provides a non-neural method for UMR graph creation from interlinear glossed text, complementing our focus on the conversion process.

Initial work by Bonn et al. (2023) and Wein and Bonn (2023) provides an analysis of the fine-grained structural distinctions between AMR and UMR, delving into key differences like tense, modality, scope, and document-level dependencies in monolingual and multilingual settings. Building upon this foundation, Bonn et al. (2023) offer a specific road-map for bridging the gap. This paper meticulously details the structural differences between AMR and UMR representation techniques for semantic categories, highlighting crucial aspects like tense, modality, scope, and document-level temporal relations. It also sheds light on the fundamental differences in graph structure, with AMR relying on predicate-argument structures and UMR accommodating polysynthetic and agglutinating languages with more complex morphologies.

By leveraging these insights, our work aims to tackle a key piece of this conversion puzzle. We focus on applying a neuro-symbolic method to address the data scarcity challenge by leveraging domain knowledge and neural learning to facilitate robust and accurate conversion, paving the way for wider UMR adoption and enhanced cross-lingual semantic analysis capabilities. We focus specifically on the non-deterministic roleset changes, contributing to a more robust and comprehensive conversion process.

This work proposes a novel data augmentation approach specifically designed for AMR to UMR role conversion. Our model builds upon the concept of constrained indirect supervision (Wang and Poon, 2018), and combines noisy examples with interdependent label constraints to address data scarcity. Several studies have explored data augmentation for NLP tasks in low supervision settings, including active learning (Quteineh et al., 2020) and rule-based approaches (Zhao et al., 2021). We leverage active learning principles by selecting informative AMR graphs containing split roles like *:destination*, *:cause*, and *:source*. Then, we incorporate animacy parsing, which is crucial for role determination, and derive logic rules from UMR guidelines to generate additional training examples and guide the neural network towards accurate role mappings. This combined approach efficiently utilizes limited labeled data and addresses the chal-

Documents	Number of Sentences	Number of Aligned Split Roles
Lindsay Text	2	1
Phillippines Landslide News Text	28	36
Putin News Text	12	15
Edmund Pope News Text	9	9
Pear Story	141	30
Total	**192**	**91**

Table 1: Parallel AMR-UMR documents with their sentence counts, and the number of split roles

lenges of low supervision settings.

3. Data

The available published parallel AMR-UMR data we utilized consists of five documents (Bonn et al., 2024), all in English, as detailed with sentence counts in Table 1. These documents vary in length and sentence complexity, ranging from short examples, like the *Lindsay Text*, to longer news stories with complex sentence structures, such as the *Philippines Landslides News Text*. In the roughly 200 AMR/UMR graphs, only about 100 split-rolesets are available for analysis.

Although prior literature has indicated the expected split role mapping (Bonn et al., 2023), initial tests have shown that this mapping is not fully captured in the data. Figure 2 shows the counts of AMR and UMR roles from all data overlaying the expected mapping. The data does not reflect a clean 1:many mapping relation. For example, the AMR role *:destination* should split into *:goal* and *:recipient*. The AMR documents consist of 2 instances of the *:destination* role but the UMR documents contain 3 instances of a *:goal* role, meaning that a different AMR role turned into the UMR role *:goal*. This does not reflect the clean splits shown in (Bonn et al., 2023). This analysis highlights the need for a more nuanced approach to role conversion.

3.1. Alignment

To gain deeper insights, we perform partial alignment of AMR and UMR graphs, focusing on the role edges. A partial alignment is possible because the information being captured is just the split-role in question. The meaning representation graphs are directed, node and edge-labeled graphs. Each edge is a semantic relation or role that connects one concept node (the head node) to another concept node (the tail node). In our data, of the 106 AMR roles that we explore, 90 have their head and tail nodes aligned to corresponding UMR graph nodes, and 70 have a matching edge in the UMR graph. Changes in UMR guidelines and structural differences between the graphs explain most misalignments[4].

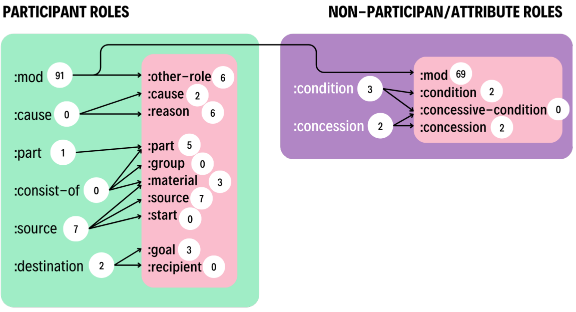


Figure 2: Split role mapping from AMR to UMR with counts from the data

UMR Label	Gold-Standard	Silver-Standard
:group	109	1
:source	90	1168
:goal	59	18
:part	59	94
:mod	58	0
:cause	53	4921
:reason	46	1419
:material	43	176
:start	34	552
:condition	16	2286
:recipient	12	0
:Cause-of	4	0
:other-role	3	0
:Material-of	1	0
Total	**587**	**10635**

Table 2: Counts of UMR Roles in gold-standard data (labels created by human annotators) and silver-star data (labels generated by Rules Model)

3.2. Data Augmentation

Because of the small amount of available parallel data, we use data augmentation to produce more **gold-standard evaluation data** and a large amount of **silver-standard training data**. The resulting dataset statistics are reported in Table 2.

Gold Standard Data To produce additional evaluation data, we employ task-specific data augmentation, leveraging elements of active and curriculum learning techniques (Jafarpour et al., 2021). This approach efficiently utilizes labeled data by manually converting AMR graphs containing split-roles

[4] UMR Guidelines: https://github.com/umr4nlp/umr-guidelines/blob/master/guidelines.md

to provide the most informative samples for training. We converted 40 additional AMR graphs to UMR graphs, preferring graphs that include roles from the less represented splits in the data. Specifically, we focus on the AMR roles of *:destination*, *:cause*, *:consist-of*, and *:source*. The sentences were chosen from the AMR data and guidelines.[5]

In a second augmentation step, we run additional data from the published AMR dataset through the rule-based model detailed in section 4.2. For a targeted set of AMR graphs, an annotator assessed the UMR role assigned by the rule-based model and corrected those labels as needed. This approach yielded 470 additional gold-standard split-role labels.

Silver Standard Data To generate additional, automatically-labeled, and thus noisy, training data, we next run the rest of the non-parallel AMR data through the rule-based model. This data is comprised of around 70,000 additional rolesets. Nearly 60,000 of these are labeled with the *:mod* role, which is both over-represented in the data and nearly always maps to the same role in UMR. For this reason, we exclude *:mod* from the silver-standard training data. The remaining 10,635 rolesets are used as silver-standard training data in Experiment 2 (see section 5).

4. Methodology

Within the broader task of automated AMR-to-UMR graph conversion, we address the specific challenge of non-deterministic role changes, reducing the need for intervention from expert human annotators. This section describes our methodologies for incorporating animacy information and logical rules into a neural architecture.

System Overview We first extract detailed information about roles from both AMR and UMR graphs, including roleset labels, head and tail entities, and their connection to the original sentence and graph context, as explained in section 3.1. An animacy recognition module, detailed in section 4.1, then determines the animacy of each role's tail, as animacy plays a crucial role in UMR role determination.

Next, all of the extracted information serves as input for a rule-based role-labeling component. The rules were formulated manually through our investigation of the logic detailed in the UMR guidelines, and they rely heavily on animacy information, as explained in section 4.2. The rule-based module outputs potential split-role conversions for the AMR role, along with their initial weights, which are determined by analyzing the frequency of role splits based on the implemented rules and the distribution of such splits within the initial UMR published data.

Final role predictions are done by three different models (section 4.3): a baseline rules-only model, a baseline neural network, and a hybrid model combining rules with neural learning. Each model receives the extracted role information, animacy data, and initial weights, utilizing them in different ways to predict the most likely UMR role.

4.1. Animacy parsing

Accurate animacy depiction is crucial for the rule-based decision-making module of our framework. According to the UMR guidelines, certain rolesets should only be used for animate or inanimate entities. Therefore, we test several existing animacy parsers and named entity recognizers (NERs), in addition to using information found within the AMR graph, to synthesize an animacy recognition module tailored to our framework from four components. Certain split-roles, such as *:mod*, were excluded from animacy parsing. Roles such as *:mod* do not need animacy information in order to determine their split, so they were excluded in order to make the model run more efficiently.

1. BERT-Finetuned-Animacy: The first component of the animacy parser is a BERT-finetuned-animacy model (Tobin, 2022). This model takes the sentence to be converted as input, and outputs entities it identifies as persons or animals.

2. BERT-NER: Next, we include a popular NER model, again taking the sentence as input and outputting labeled named entities (person, named organization, named place, and misc) (Lim, 2023).

3. Pronouns: Next, we search for any pronouns within the sentence. While pronouns such as "I", "you", and "she" are not always necessarily animate, they are enough of a proxy for animacy in our data that we chose to include them in the animacy distinction, marking them as "person" roles.

4. AMR Named Entities: The AMR guidelines define various named entities (NEs) in the tails of many role instances. We manually assign animacy labels ("animate" or "inanimate") to each of the NE types. However, akin to the limitations of using pronouns for animacy prediction, this approach overclassifies entities as animate. Overprediction of the "animate" label helps to balance against the animacy parser's tendency to default to "inanimate".

[5] AMR guidelines: https://github.com/amrisi/amr-guidelines/blob/master/amr.md#reification.

Even with over-prediction, the model only produced animate tags as opposed to inanimate tags 2.88% of the time on the full augmented dataset.

Animacy Integration We use the outputs of the various components to make a binary animate/inanimate distinction for each role. First, we check the tail of each role against the items returned as animate. If there is no match for the tail, we next check for a child role, in cases where the tail has a role sense (e.g., *believe-01, leave-14, survive-02*). If there are no matches between the sentence and the outputs of the animacy parser, we treat the role as inanimate.

4.2. Split-Role Rules

Both for prediction and for creating silver-standard data, we encode a set of logical rules capturing tendencies in the mapping of AMR roles to UMR roles. This section details the rules, organized according to original AMR roleset. The rules were created manually as detailed in each section through study of the AMR and UMR guidelines, as well as by referencing UMR examples in our training dataset. In the future, we see the potential to create more rules-based modules to help with the conversion of other split-roles.

Destination Roleset Bonn et al. (2023) substantiate that the AMR *:destination* role splits into the UMR roles *:goal* and *:recipient*. The UMR guidelines additionally specify information about the animacy of certain rolesets. For the *:recipient* role, the UMR guidelines define *:recipient* as an "animate entity that gains possession (or at least temporary control) of another entity". The *:goal* role does not have specified animacy. The resulting rule is that if the AMR *:destination* role is inanimate, the UMR role must be *:goal*. If the AMR *:destination* role is animate, the UMR role may be *:recipient* or *:goal*.

Cause Roleset The second rule addresses the AMR *:cause* role. Similar to the *:destination* role, this role is split using animacy into *:cause* and *:reason*. The UMR guidelines note that the UMR *:cause* role is an "inanimate entity that causes the action to happen." The resulting rule is that if the tail of the AMR *:cause* role is animate then the UMR role must be *:reason*. Otherwise, if it is inanimate, the UMR role may be *:reason* or *:cause*.

Source Roleset The third rule addresses the AMR role *:source*, as illustrated in Figure 3. The *:source* role may split into three different UMR roles: *:source*, *:start*, *:material*. The UMR guidelines give helpful information about animacy for these roles, as well as guidance on the parent role of the instance. For instance, the guidelines provide that the tail of the *:source* roled must be animate. We encode this information by first checking if the tail roleset of the AMR role is animate. If so, the UMR role is set to *:source* since the other roles are generally inanimate. Then, we check if the parent node of the AMR *:source* is *:theme*, as the UMR guidelines specify that *:source* is the "entity from which the *:theme* detaches". In this case the UMR role chosen is *:source*.

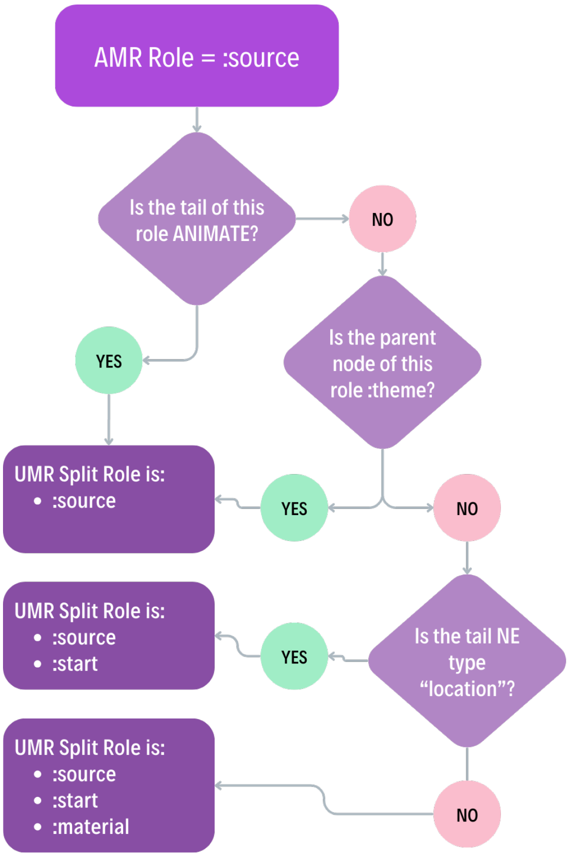


Figure 3: Animacy logic rule for UMR :source, :start, and :material roles from AMR :source role

Next, the animacy and NE info from the animacy parser is checked to see if it contains a location. If so, the UMR role chosen is *:source* or *:start*. Finally, if the tail roleset of *:source* is inanimate, then the role is either *:source*, *:start*, or *:material*. We obtain initial probabilities for these rule assignments using the distributions observed in the gold-standard UMR graphs (e.g., 0.6 for *:source*, 0.3 for *:start*, and 0.1 for *:material*).

Consist-of Roleset This rule relies on animacy to determine the AMR *:consist-of* role-split. The UMR role *:group* is the only animate role and will always be chosen if the tail AMR roleset is animate. Otherwise, the UMR roles *:group*, *:part*, or *:material* may be the correct split-role choice.

Additional Rolesets The roles *:part* and *:condition* deterministically split into the UMR roles with identical names in English.

The final rule addresses the AMR role *:mod*. This role only rarely maps into the UMR role *:other-role*. Due to the lack of clear rules for this role, we rely on the neural methods to improve prediction accuracy. Initial weights favor *:mod* over *:other-role*.

4.3. Models

We investigate three different models: one using rules alone, one simple neural architecture with no rules, and one combined model.

Rules-only model: For each AMR role, there are 1-3 possible UMR roles. The possible roles are determined by the previously-defined rules, given the AMR role, its predicted animacy, and the AMR graph information. When there is not enough information for the rules to narrow down to just one possible role, the model randomly selects a role label according to the probability distributions seen for that AMR role in the gold-standard parallel UMR data.

Neural Network: Our neural network implementation is a three layer feed-forward neural network. It takes as input the Sentence-BERT embedding of the sentence (Reimers and Gurevych, 2019), concatenated with a feature representing the source AMR role. Although it does not use the animacy rules to influence training, we incorporate external knowledge in constraining the outputs to be only what is possible for the AMR-role to convert to given the UMR guidelines. (For instance, *:destination* can only be converted to *:goal* or *:recipient*). The constraints can be viewed in Figure 2. To train the classifier, we use the cross-entropy loss.

NN with Rule Information: We opt for a simple implementation of a neural network that has access to the rule information in an attempt to leverage the logic of the rules with the predictive power of a neural network. We incorporate the rule information in two ways: 1) We concatenate the probability distribution of the possible roles provided by the rules to the sentence embedding and the AMR role, as the input to the NNet, and 2) We add an additional layer to combine the output (argmax) of the neural network and the rules as: $w_1 * output_{NN} + w_2 * output_{rules}$, where w_1 and w_2 correspond to the trainable parameters of the additional layer. The classifier is then trained end-to-end using the cross-entropy loss.

5. Experiments

We evaluate our three models in two different settings: one training only on gold-standard data, and one adding noisily-labeled (silver-standard) data to the training sets. To better understand how performance is influenced by the difficulty of the particular decision, we categorize the roles into four bins. The first bin, "easy," includes roles with deterministic picks for the English data. The second bin, "medium," consists of roles for which accurate animacy information should lead to accurate role determination. The third bin, "medium/hard," includes roles with more than one choice within each animacy category. Finally, roles in the "hard" bin do not have the benefit of guidance of animacy and have multiple split-roles they could fall into. See Table 5 in appendix for further details.

Experimental Settings For all experiments, we use stratified 5-fold cross-validation and report average results. In each iteration, we use 4 folds for training and 1 for testing. The rules-only model involves no training, so the results shown are based on the predictions for each fold's test set. Results are averaged over 5 runs. Experiment 1, our low-data experiment, uses only the gold-standard data for training and testing. In experiment 2, we use the same folds as experiment 1, now augmenting every fold's training data with the 10,635 silver-standard data points (sec. 3.2). With these settings, we evaluate only on gold-standard data, always include some amount of gold data in training, and ensure comparability across experiments. Our main evaluation measure is macro F1. We also report the weighted F1, which takes into account the label distribution in the test data.

For training, both neural models use a learning rate of 0.001 and train over 50 epochs.

5.1. Experiment 1 - Low data

In this experiment, only gold-standard data is used for training, with an average of just 470 training instances per fold. Per-fold performance is shown in Fig. 4. The Rules model and NN_Rules model perform similarly, and the NN struggles, with high variation across folds. We aggregate the five test sets to evaluate per-class performance as reported in Table 3. The NN_Rules model has the highest F1 score in 7 classes, more than either the Rules model or the NN model. In a small data setting like this, it is not unexpected to see the NN struggle to perform well.

5.2. Experiment 2 - Weak supervision

This experiment combines the gold-standard and silver-standard sets for training, allowing for more

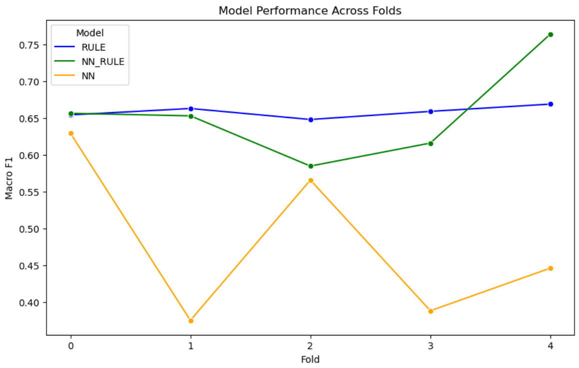


Figure 4: Experiment 1: Macro F1 performance of the three models across 5 folds.

Difficulty	Label	NN	NN_RULE	RULE	support
easy	:condition	1.000	1.000	1.000	16
	:mod	0.838	**0.966**	0.956	58
	:part	0.622	**0.778**	0.775	59
medium	:goal	0.894	0.873	**0.950**	59
	:other-role	0.118	**0.500**	0.000	3
medium/hard	:group	0.724	0.815	**0.913**	109
	:reason	0.000	0.407	**0.653**	46
	:source	0.610	**0.802**	0.689	90
hard	:Cause-of	0.240	**0.857**	0.000	4
	:Material-of	0.000	0.000	0.000	1
	:cause	0.637	**0.743**	0.737	53
	:material	0.419	**0.582**	0.552	43
	:recipient	0.000	0.222	**0.840**	12
	:start	**0.379**	0.351	0.265	34
	macro avg F1	0.463	**0.635**	0.595	
	weighted avg F1	0.603	0.738	**0.761**	587

Table 3: Per-class F1-scores from experiment 1, arranged by the difficulty of the split decision. Bolded values are the highest in each class.

training data in a low-supervision setting. The macro-F1 performance of all of the models across the folds can be seen in Figure 5. Once again, the Rules model and NN_Rules model perform similarly across the folds, and although the basic NN shows reduced performance, there is more consistency across the folds. Per-class performance

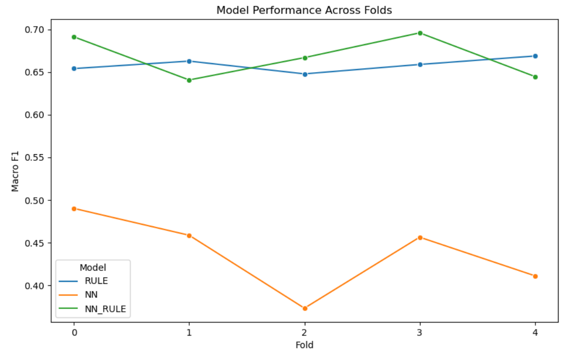


Figure 5: Experiment 2: Macro F1 performance of the three models across 5 folds.

Difficulty	Label	NN	NN_RULE	RULE	support
easy	:condition	1.000	1.000	1.000	16
	:mod	**0.958**	0.948	0.956	58
	:part	0.652	**0.775**	**0.775**	59
medium	:goal	0.894	**0.949**	**0.949**	59
	:other-role	0.000	**0.333**	0.000	3
medium/hard	:group	0.607	**0.936**	0.913	109
	:reason	0.000	0.600	**0.653**	46
	:source	**0.754**	0.685	0.689	90
hard	:Cause-of	**0.114**	0.000	0.000	4
	:Material-of	0.000	0.000	0.000	1
	:cause	0.640	**0.758**	0.737	53
	:material	0.182	**0.645**	0.551	43
	:recipient	0.000	**0.846**	0.840	12
	:start	0.207	0.212	**0.265**	34
	macro avg F1	0.429	**0.621**	0.595	
	weighted avg F1	0.589	**0.767**	0.761	587

Table 4: Per-class F1-scores from experiment 2, arranged by the difficulty of the split decision. Bolded values are the highest in each class.

(aggregating all folds) is reported in Table 4. In this experiment, the NN_Rules model scores higher than the Rules model on summary statistics, and it achieves the highest score in more classes.

6. Discussion

In this section, we discuss our experimental results, perform a detailed error analysis, and outline directions for future work.

6.1. Experimental Results

In our experiments, we observed that the vanilla neural network struggled to obtain good performance across all configurations. While we saw improvements in the augmented data scenario, the amount and the quality of the supervision was not enough to outperform a simple rule based model. The rule-based model, on the other hand, delivered good average performance across all configurations. This is in line with the highly constrained nature of our task. However, we saw that the NN augmented with rules scored the highest in average when exposed to additional training data. This suggests that hybrid models are a good alternative in weak supervision scenarios, where the neural network can take advantage of the augmented data, while the structured knowledge can help guide the model towards valid answers. For all models, performance was higher for the easy cases than for the hard cases.

To showcase the impact of having access to high-quality annotations, we will consider the *:cause* role. Within the silver-star data, over 40% of the roles were labeled *:cause*. The Rules model performs well on *:cause* with an F1-score of 0.737. Having a large number of high quality labels for training is reflected in the performance of both the NN and the NN with Rules in the *:cause* role. Both models perform better for this class in experiment 2 than they did in experiment 1. Conversely, when examining the performance for the class *:start*, which has over 500 labeled instances in the silver-standard set, we see that the Rules model's low performance (0.314) adversely affects the ability of the neural methods to predict this class.

6.2. Error Analysis

In this section we discuss specific types of errors made by various models. Some additional examples appear in Table 6, in the appendix.

One prevalent error made by the Rules model comes from adhering to the initial label distributions, as this model has no ability to take context into account. For example, in the sentence *"I saw a cloud of dust"*, the Rules model maps the AMR *:consist-of* role to the UMR role *:group*, for the tail *dust*. In contrast, the NN_Rules model correctly identifies that *:consist-of* should be mapped to the UMR role *:material*. The NN_Rules model leverages learned information to make more informed predictions. All roles in the "medium/hard" category are subject to this type of error.

Another error class occurs when the Rules model fails to make a correct prediction due to inaccuracies in animacy determination. For instance, in the sentence *"A letter from the victim's family,"* the tail role *"family"* was incorrectly parsed as *inanimate*, leading to an incorrect choice of role label. However, the NN_Rules model is not affected by this incorrect animacy parsing, demonstrating better performance in "medium" difficulty scenarios, where correct animacy parsing is needed for the rules to make accurate labeling decisions.

All models encounter difficulty with inverse participant roles such as *:Cause-of* and *:Material-of*. Inverse participant roles, as described in the UMR Guidelines, involve moves like annotating events as modifiers or referring expressions, requiring more complex graph modifications than we currently handle. They are also very infrequent in the data. These rolesets are part of the "hard" category.

Despite similar overall performance to the Rules model, the NN_Rules model shows improvements for roles in "hard," "medium-hard," and "medium" difficulty scenarios. This result highlights the potential of combining symbolic and neural approaches for improved AMR-to-UMR conversion.

6.3. Future Work

Animacy Given the strong influence of the animacy parser, this is an obvious avenue for improvement. Recent studies (e.g. Hanna et al., 2023) highlight challenges for language models in handling subtle shifts in animacy cues within text. While our current approach incorporates animacy information from UMR guidelines, including context-dependent animacy shifts for typical entities, it is still under development in terms of capturing the full spectrum of animacy variations. Additionally, treating animacy as a binary decision might not fully capture the nuances explored in studies like Ji and Liang (2018), which propose a hierarchical spectrum of animacy even within inanimate nouns. For example, "robot" might exhibit more animacy than "chair" due to its potential for movement and agency.

Alternative Modeling Strategies Our NN_Rules model incorporates rules into the neural network in a naive way. In the near future, we intend to investigate alternatives like combining neural networks with Probabilistic Soft Logic (PSL) (Bach et al., 2017) or employing neuro-symbolic methods that leverage rules like DRAIL, a deep relational learning framework (Pacheco and Goldwasser, 2021). An improvement to our current implementation could make use of the full graph-structure of the MRs, instead of just extracting relevant edges. Additionally, different approaches to using and combining the silver-standard and gold-standard datasets could prove beneficial. For example, curating the silver-standard data to remove the labels from low-quality classes, and using a split of the gold-standard data during development, may leverage the strength of both datasets more effectively.

Expansion to other UMR Components In the future, we believe this methodology can be applied to other parts of the AMR-UMR conversion process, starting with expansion to all of the semantic roles, not just this subset of role changes. By thoughtfully constructing rules, we can potentially aid annotators throughout the entire annotation process. Graph preprocessing approaches like handling inversion and reification could prove beneficial to more complex changes.

User Study In the spirit of demonstrating the usefulness of our tool to the UMR annotator audience it is intended for, we propose an experiment evaluating its impact on annotation speed and accuracy. This experiment would involve experienced UMR annotators working on two sets of AMR graphs each:

1. Traditional: Annotators complete the conversion task without any additional information or assistance.

2. Tool-assisted: Annotators leverage our model's predicted split-role conversions alongside the AMR graphs.

By comparing annotation times and accuracy between the two groups, we can assess the potential benefits of our tool in expediting and potentially improving the UMR annotation process. This evaluation aligns with our goal of providing UMR annotators with valuable resources to streamline their workflow.

7. Conclusion

This work presented a novel, modular methodology for automated AMR-to-UMR graph conversion, with a primary focus on accurately predicting non-deterministic role changes that often require human intervention. Our approach integrates animacy parsing, logic rules, and neural learning to achieve promising accuracy.

Key contributions include introducing a modular framework for easy integration with future techniques, promoting extensibility and broader applicability. Furthermore, the incorporation of animacy information enhances decision-making in role prediction, while the fusion of structured knowledge with neural learning offers flexibility and robustness. The model's encouraging performance on the test data highlights its potential to streamline the conversion process and thus accelerate UMR adoption.

While acknowledging the promising results, we recognize limitations arising from data scarcity and the binary representation of animacy. Future work will involve expanding animacy parsing to capture richer semantic information and context-dependent nuances, potentially employing non-binary representations to improve accuracy. Additionally, user studies will be conducted to assess the impact of our methodology on UMR annotation speed and accuracy, providing valuable insights into its practical utility. Finally, we envision expanding our approach to encompass broader aspects of AMR-UMR conversion, further contributing to the advancement of cross-lingual semantic analysis and unlocking the full potential of UMR for multilingual NLP tasks.

This research demonstrates the benefits of combining symbolic and neural approaches for complex NLP tasks in data-constrained scenarios. By overcoming data scarcity challenges and facilitating accurate UMR conversion, our method paves the way for enhanced cross-lingual semantic analysis capabilities, ultimately impacting various NLP applications that rely on accurate semantic representation and understanding.

8. Limitations

Animacy, the distinction between animate and inanimate entities, plays a crucial role in determining split roles within our rule-based model. It influences the roles a referent can take on, for instance, requiring animacy for the agent role. While existing animacy classifiers like those presented in Tobin (2022); Jahan et al. (2018) exist, they can be imperfect and miss participants within sentences where animacy is nuanced or context-dependent. This limitation can lead to inaccurate role predictions in certain cases.

As well, this work faces several data-related challenges that limit the scope of model development. The limited availability of parallel AMR-UMR annotations, consisting of an extremely small dataset of only 200 graphs from five documents (see Table 1), constrained our ability to train and evaluate models effectively. Moreover, inconsistencies between expected and observed role mappings (as illustrated in Figure 2) suggest a more nuanced conversion process than a simple 1:many relationship, complicating model training and interpretation. Our current focus on sentence-level conversion also limits the applicability of our model to larger discourse contexts. And finally, data imbalances, particularly with over-represented roles like ":mod," created issues in the analysis and data augmentation steps.

9. Acknowledgements

This work is supported by a grant from the CNS Division of National Science Foundation (Award no: NSF_2213805) entitled “Building a Broad Infrastructure for Uniform Meaning Representations.” We extend our sincere gratitude to Julia Bonn for her invaluable insights, suggestions, and expertly crafted spreadsheets elucidating AMR and UMR. Special thanks are due to the participants of the CU Boulder 2023 Neuro-Symbolic Approaches to NLP class for their constructive feedback during the development of this work. Lastly, we express our appreciation to the reviewers for their helpful comments and feedback.

10. References

Stephen Bach, Matthias Broecheler, Bert Huang, and Lise Getoor. 2017. Hinge-Loss Markov Random Fields and Probabilistic Soft Logic. *Journal of Machine Learning Research (JMLR)*, 18(1):1–67.

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 *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*, pages 178–186.

Julia Bonn, Matthew Buchholz, Jayeol Chun, Andrew Cowell, William Croft, Lukas Denk, Sijia Ge, Jan Hajič, Kenneth Lai, James H. Martin, Skatje

Myers, Alexis Palmer, Martha Palmer, Claire B. Post, James Pustejovsky, Kristine Stenzel, Haibo Sun, Zdeňka Urešová, Rosa Vallejos, Jens E. L. Van Gysel, Meagan Vigus, Nianwen Xue, and Jin Zhao. 2024. Building an Infrastructure for Uniform Meaning Representations. Language Resources and Evaluation (LREC), The 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation.

Julia Bonn, Skatje Myers, Jens E. L. Van Gysel, Lukas Denk, Meagan Vigus, Jin Zhao, Andrew Cowell, William Croft, Jan Hajič, James H. Martin, Alexis Palmer, Martha Palmer, James Pustejovsky, Zdenka Urešová, Rosa Vallejos, and Nianwen Xue. 2023. Mapping AMR to UMR: Resources for adapting existing corpora for cross-lingual compatibility. In *Proceedings of the 21st International Workshop on Treebanks and Linguistic Theories (TLT, GURT/SyntaxFest 2023)*, pages 74–95, Washington, D.C. Association for Computational Linguistics.

Matthew Buchholz, Julia Bonn, Claire B. Post, Andrew Cowell, and Alexis Palmer. 2024. Bootstrapping UMR Annotations for Arapaho from Language Documentation Resources. Language Resources and Evaluation (LREC), The 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation.

Jens Van Gysel, Meagan Vigus, Jayeol Chun, Kenneth Lai, Sarah Moeller, Jiarui Yao, Timothy J. O'Gorman, Andrew Cowell, W. Bruce Croft, Chu-Ren Huang, Jan Hajic, James H. Martin, Stephan Oepen, Martha Palmer, James Pustejovsky, Rosa Vallejos, and Nianwen Xue. 2021. Designing a Uniform Meaning Representation for Natural Language Processing. *KI - Künstliche Intelligenz*, 35:343 – 360.

Michael Hanna, Yonatan Belinkov, and Sandro Pezzelle. 2023. When Language Models Fall in Love: Animacy Processing in Transformer Language Models. *arXiv preprint arXiv:2310.15004*.

Borna Jafarpour, Dawn Sepehr, and Nick Pogrebnyakov. 2021. Active Curriculum Learning. In *Proceedings of the First Workshop on Interactive Learning for Natural Language Processing*, pages 40–45, Online. Association for Computational Linguistics.

Labiba Jahan, Geeticka Chauhan, and Mark A Finlayson. 2018. A New Approach to Animacy Detection. *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1–12.

Jie Ji and Maocheng Liang. 2018. An animacy hierarchy within inanimate nouns: English corpus evidence from a prototypical perspective. *Lingua*, 205:71–89.

David S. Lim. 2023. Model: Bert Base NER.

Maria Leonor Pacheco and Dan Goldwasser. 2021. Modeling Content and Context with Deep Relational Learning. *Transactions of the Association for Computational Linguistics*, 9:100–119.

Husam Quteineh, Spyridon Samothrakis, and Richard Sutcliffe. 2020. Textual data augmentation for efficient active learning on tiny datasets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7400–7410.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

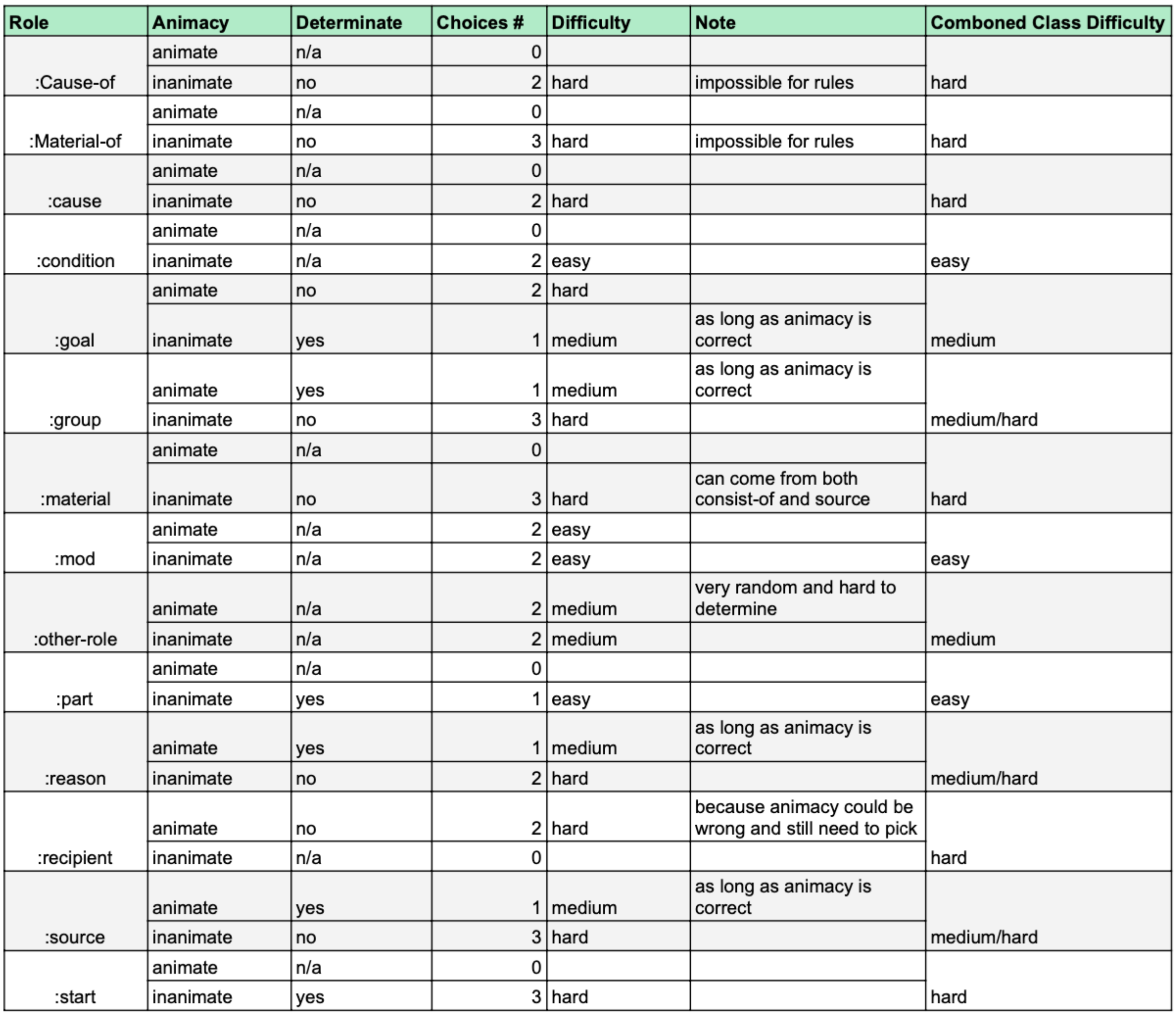
Andrew Tobin. 2022. Model: Bert Finetuned Animacy.

Hai Wang and Hoifung Poon. 2018. Deep Probabilistic Logic: A Unifying Framework for Indirect Supervision. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Shira Wein and Julia Bonn. 2023. Comparing UMR and Cross-lingual Adaptations of AMR. *Proceedings of the 4th International Workshop on Designing Meaning Representations*.

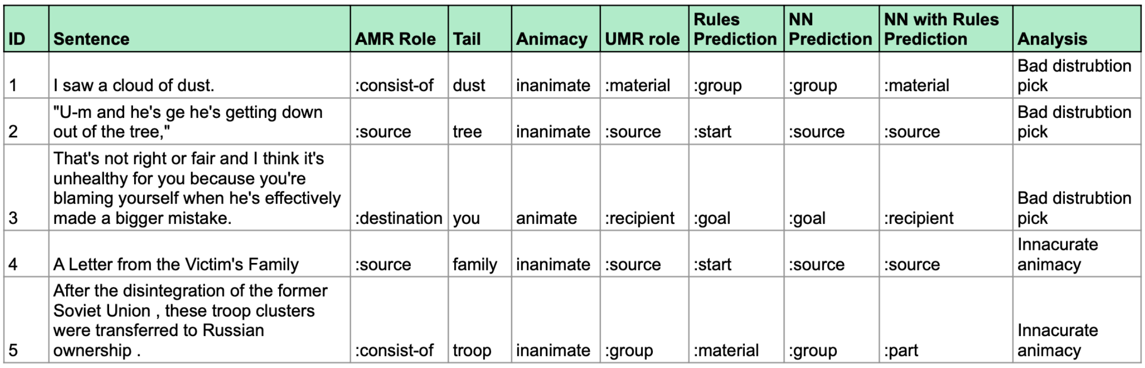
Xinyan Zhao, Haibo Ding, and Zhe Feng. 2021. GLaRA: Graph-based labeling rule augmentation for weakly supervised named entity recognition. *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3636–3649.

A. Appendix



Role	Animacy	Determinate	Choices #	Difficulty	Note	Comboned Class Difficulty
:Cause-of	animate	n/a	0			hard
	inanimate	no	2	hard	impossible for rules	
:Material-of	animate	n/a	0			hard
	inanimate	no	3	hard	impossible for rules	
:cause	animate	n/a	0			hard
	inanimate	no	2	hard		
:condition	animate	n/a	0			easy
	inanimate	n/a	2	easy		
:goal	animate	no	2	hard		medium
	inanimate	yes	1	medium	as long as animacy is correct	
:group	animate	yes	1	medium	as long as animacy is correct	medium/hard
	inanimate	no	3	hard		
:material	animate	n/a	0			hard
	inanimate	no	3	hard	can come from both consist-of and source	
:mod	animate	n/a	2	easy		easy
	inanimate	n/a	2	easy		
:other-role	animate	n/a	2	medium	very random and hard to determine	medium
	inanimate	n/a	2	medium		
:part	animate	n/a	0			easy
	inanimate	yes	1	easy		
:reason	animate	yes	1	medium	as long as animacy is correct	medium/hard
	inanimate	no	2	hard		
:recipient	animate	no	2	hard	because animacy could be wrong and still need to pick	hard
	inanimate	n/a	0			
:source	animate	yes	1	medium	as long as animacy is correct	medium/hard
	inanimate	no	3	hard		
:start	animate	n/a	0			hard
	inanimate	yes	3	hard		

Table 5: Difficulty of the decision of each role, reflected in the number of possible roles the model must choose from, even with the animacy information and the rules.



ID	Sentence	AMR Role	Tail	Animacy	UMR role	Rules Prediction	NN Prediction	NN with Rules Prediction	Analysis
1	I saw a cloud of dust.	:consist-of	dust	inanimate	:material	:group	:group	:material	Bad distrubtion pick
2	"U-m and he's ge he's getting down out of the tree,"	:source	tree	inanimate	:source	:start	:source	:source	Bad distrubtion pick
3	That's not right or fair and I think it's unhealthy for you because you're blaming yourself when he's effectively made a bigger mistake.	:destination	you	animate	:recipient	:goal	:goal	:recipient	Bad distrubtion pick
4	A Letter from the Victim's Family	:source	family	inanimate	:source	:start	:source	:source	Innacurate animacy
5	After the disintegration of the former Soviet Union , these troop clusters were transferred to Russian ownership .	:consist-of	troop	inanimate	:group	:material	:group	:part	Innacurate animacy

Table 6: Error analysis of several common error types ran from Experiment 2.

The Relative Clauses AMR Parsers Hate Most

Xiulin Yang, Nathan Schneider
Georgetown University
Washington, DC, USA
{xy236, nathan.schneider}@georgetown.edu

Abstract

This paper evaluates how well English Abstract Meaning Representation parsers process an important and frequent kind of Long-Distance Dependency construction, namely, relative clauses (RCs). On two syntactically parsed datasets, we evaluate five AMR parsers at recovering the semantic reentrancies triggered by different syntactic subtypes of relative clauses. Our findings reveal a general difficulty among parsers at predicting such reentrancies, with recall below 64% on the EWT corpus. The sequence-to-sequence models (regardless of whether structural biases were included in training) outperform the compositional model. An analysis by relative clause subtype shows that passive subject RCs are the easiest, and oblique and reduced RCs the most challenging, for AMR parsers.

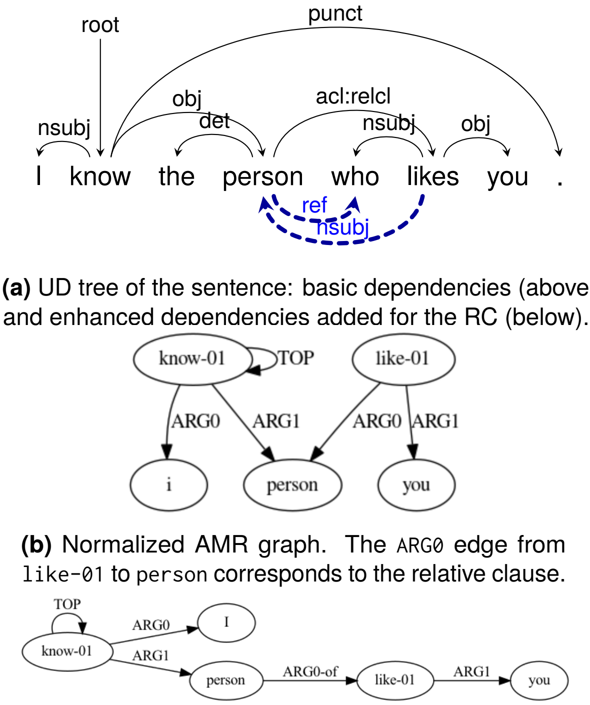
Keywords: AMR, Relative Clause, Semantic Parsing

1. Introduction

Abstract Meaning Representation (AMR; Banarescu et al., 2013) has emerged as a mainstream framework in semantic parsing tasks. Recent advancements in AMR parsers have led to significant achievements, with scores over 0.85 (Lee et al., 2022) in Smatch (Cai and Knight, 2013). However, relying solely on overall F-scores does not fully reveal a parser's performance across different linguistic phenomena, leaving areas for improvement and potential problems unclear.

In semantic parsing tasks, previous research has shown that sequence-to-sequence (seq2seq) models are good at abstracting away from surface variation in how meanings are expressed (Shaw et al., 2021). However, seq2seq models that process symbolic structures as mere strings face challenges in compositional generalization, such as the ability to process recursion, compared to models designed to be sensitive to the structure (Yao and Koller, 2022; Li et al., 2023; Shaw et al., 2021). This raises the possibility that such "structure-awareness" in the design of semantic parsers may be valuable for complex constructions generally.

In this paper, we focus on evaluating AMR parsers on English relative clauses, a frequent Long-Distance Dependency (LDD) construction. LDD refers to the linguistic phenomenon that two elements in a sentence, though not adjacent to each other, are still syntactically/semantically constrained. As a typical example of LDD, RCs are a popular topic of computational linguistic study (e.g., Davis and van Schijndel, 2020; Ravfogel et al., 2021). Compared with non-LDD constructions, RCs are structurally complex and may give rise to semantic ambiguities, so we assume they will be challenging for parsers. Figure 1 shows syntactic dependencies and the semantic AMR graph for an example RC.



(a) UD tree of the sentence: basic dependencies (above) and enhanced dependencies added for the RC (below).

(b) Normalized AMR graph. The `ARG0` edge from `like-01` to `person` corresponds to the relative clause.

(c) Canonical AMR graph. The `ARG0-of` edge corresponds to the relative clause.

Figure 1: UD and AMR representations for the sentence containing a subject relative clause *I know the person who likes you.* Converting the canonical (annotated/parsed) graph into the normalized one entails inverting the `-of` edges, causing nodes to be reentrant (have multiple parents).

In our evaluation we examine two types of AMR parsers: structure-aware and structure-unaware models. **Structure-unaware** models, as defined herein, process input purely as sequential strings; algorithms for learning and decoding are indifferent to any notation within these strings that represents sentence structure. Conversely, **structure-aware** models are designed to take into account structural information, thereby enabling a more nuanced understanding of the input data's inherent syntactic and semantic properties.

We ask: **How well can AMR parsers capture the long-distance predicate-argument dependencies in RCs?** To answer this question, we normalize edges that contain `-of` by inverting the source node and the target node, and then evaluate parsers by measuring recall of the reentrancies introduced by RCs in two datasets: Universal Dependencies English Web Treebank (EWT; Silveira et al., 2014) and Controlled RCs (CRC; Prasad et al., 2019). Our investigation engages with the following subquestions:

- Does structure-awareness help the models to parse RCs in EWT and CRC?
- Which types of RC are most challenging and why?

Our contributions include:

- A fine-grained method to classify RCs and annotate Enhanced Universal Dependencies (EUD) in reduced RCs automatically.
- A systematic comparison of five AMR parsers, focusing specifically on their accuracy in parsing reentrancies introduced by RCs, along with an analysis of the underlying reasons for their performance differences.

This paper begins with an overview of RCs and reentrancies in AMR parsing (§2), followed by an introduction to the dataset, classification algorithm, and models in §3. §4 presents and discusses the results of our evaluation. The conclusion and suggestions for future research directions are presented in §5.[1]

2. Background & Related Work

2.1. Relative Clauses

In a canonical RC, a noun is modified by a clause and is understood to fulfill a grammatical function within that clause. The modified noun is the *head* of the RC. Some RCs have a *relative pronoun* like *which* or *that*. When the relative pronoun is omitted, the clause is termed a *reduced RC*; when the relative pronoun is present, along with a full clause structure, it is termed a *full RC*. According to the NP accessibility principle (Keenan and Comrie, 1977), English allows relativization on all grammatical functions. In the present study, we focus on four types of full RCs and two of their reduced counterparts:

- **Subject RC**: the relative pronoun functions as the subject of the active voice clause, as in: *He is the person **who stole my book**.*
- **Object RC**: the relative pronoun functions as the object of the clause: *He is the person **that you like**.*
- **Oblique RC**: the relative pronoun functions as an oblique within the RC: *He is the person **from whom I borrowed the book**.* All PPs attaching to verbs/adjectives are considered obliques within the UD framework, which does not distinguish oblique arguments vs. adjuncts.
- **Passive RC**: the RC is a passive clause whose subject is relativized: *He is the person **who is accused of stealing my book**.*
- **Reduced Object RC**: there is no relativizer but the head noun is understood to function as the object of the clause: *He is the person **you like**.*
- **Reduced Oblique RC**: there is no relativizer but the head noun is understood to function as the oblique of the clause: *He is the person **I borrowed the book from**.*

These are not the only kinds of RCs: there are also free relatives (e.g., *I heard what **you said***), possessive RCs (e.g., *I like the girl **whose dress is blue***), and reduced subject RCs (e.g., *I met the person **you mentioned __ finished all the work this week***; for clarity in this example, we indicate the site of the gap, i.e. where the noun would go were it not relativized).[2] However, as these are relatively rare in our dataset, our experiments are focused on the six major RC types listed above.

2.2. RCs in UD

For the present study, we use the framework of Universal Dependencies (UD, specifically UDv2; Nivre et al., 2020; de Marneffe et al., 2021), a syntactic annotation framework consisting of bilexical dependencies. UD defines a shallow dependency tree known as the *basic* tree, optionally complemented with an *enhanced* graph that adds deeper dependencies for several constructions.

The basic tree plus edges specific to the enhanced graph are illustrated in Figure 1a for a sentence with a subject RC. In the UD framework, (most) English RCs are considered a subtype of adnominal clause. The predicate of the RC attaches to the head noun with the `acl:relcl` dependency

[1] Our code and data can be found at `https://github.com/xiulinyang/relative-amr-eval`

[2] Adnominal participial clauses (*the sheep **eaten by wolves***, *the wolves **eating the sheep***) are considered RCs in some frameworks, but not in English UD (`https://universaldependencies.org/en/dep/acl-relcl.html`). There are also adverbial clauses analyzed as RCs in UD (`advcl:relcl`), e.g. in cleft sentences: *It was Booth **who shot Lincoln***. These are not very frequent in our data and we exclude them from our analysis.

relation. When a relative pronoun exists, in the basic tree, it attaches inside the RC with the relativized dependency relation. In the enhanced UD (EUD) representation, the head noun acquires the grammatical function within the RC, and the relative pronoun (if present) attaches to the head noun via a `ref` edge in lieu of its basic function.

2.3. Fine-grained AMR Evaluation

Recognizing that overall F-scores do not tell the full story of parser behavior, researchers have sought to provide a finer-grained picture of the performance of AMR parsers. Damonte et al. (2017) report the results of a wide range of general features of AMRs such as reentrancies, negative polarity, and wikification. To evaluate reentrancies, they normalize the edges in AMR so that RCs also introduce reentrancies. Our evaluation on AMR 3.0 data adopts their approach.

Szubert et al. (2020) provided a detailed analysis of reentrancies in AMR 2.0 caused by different syntactic, semantic, or pragmatic factors. They developed a set of heuristics to detect causes of reentrancies for parser evaluation. However, they focus on reentrancies in the canonical form of the AMR, whereas RCs are only reentrant in the inverse-normalized form (Figure 1), so they exclude RCs from their evaluation (Szubert et al., 2020, p. 2201).

The GrAPES benchmark (Groschwitz et al., 2023) is designed to test AMR parsers against nine specific challenging categories, which include structural generalization and syntactic as well as semantic reentrancies, among others. The dataset includes 130 RCs in a more challenging setting where sentences contain recursive RCs with optional coreference. Groschwitz et al. test three AMR parsers, all of which attain very low exact match scores ranging from 0% to 17% on these recursive RCs.

Our paper contributes to this literature by taking a deep dive on RCs, with an extensive comparison of AMR parsers across more than 1,400 corpus examples and 1,400 synthetic instances of RCs.

2.4. Probing Language Models using RCs

A few studies have used RCs to probe syntactic structures represented in language models (LMs) (e.g., Davis and van Schijndel, 2020; Mosbach et al., 2020; Prasad et al., 2019). They use either synthetic or naturalistic data to probe if the LM represents certain linguistic features or bias. For example, Davis and van Schijndel (2020) use English and Spanish RCs to examine the linguistic bias of RNN LM on the high/low attachment of RCs when trained with only synthetic or real multilingual corpus data. They found that models trained on synthetic data could learn to attach RCs both high and low in the sentence structure. However, when trained on real-world, multilingual corpus data, the models tended to favor low attachment, similar to the pattern seen in English, even though this preference is not common globally across languages. Following Kim et al. (2019); Warstadt and Bowman (2019), Mosbach et al. (2020) examined 3 pre-trained masked language models (BERT, RoBERTa, and ALBERT) on sentence-level syntactic and semantic understanding. They found that all models show high performance in parsing syntactic information but fail to predict the masked relative pronoun using context and semantic knowledge.

Dataset	# sents	# tokens
EWT	1,449	26.5
CRC	1,400	13.7
AMR 3.0	259	29.1

Table 1: Number of sentences containing RCs in the datasets and the mean sentence length

3. Method

3.1. Data

In our experiments, three datasets are used. The statistics of the dataset are reported in Table 1.

EWT UD treebank (henceforth EWT) The data we use is the train split from the Universal Dependencies English Web Treebank (Silveira et al., 2014). The original English Web Treebank contains constituency trees for diverse web text genres including weblogs, newsgroups, emails, reviews, and Yahoo! answers (Bies et al., 2012). It was then incorporated into the Universal Dependencies project; we use the dependency trees for this project.[3] Opting for the training split allows for a more extensive set of examples for evaluation. A thorough review has confirmed that there is no content overlap between EWT and the AMR 3.0 dataset (on which AMR parsers were trained).

Controlled RCs (henceforth CRC) The CRC dataset is adopted from (Prasad et al., 2019). It contains 7 types of clauses with controlled vocabulary and syntactic structures, which have been artificially generated to ensure balance across constructions and avoid potential confounds like length in comparing parser performance. We employed the four types of RCs in the dataset: subject RC, object RC, reduced object RC, and passive RC. Every category contains 350 examples.

[3] `https://github.com/UniversalDependencies/UD_English-EWT/`, specifically the dev branch as of Jan. 22, 2024, which contains changes beyond the UD 2.13 release

AMR 3.0 We report standard AMR parsing metrics on the test split of the AMR 3.0 release (Knight et al., 2021), which consists of gold AMR annotations from a variety of genres, including especially news and online discussion forums. We also report reentrancy recall on subject relative clauses.

3.2. RC classification

To have a fine-grained evaluation, we need to classify the sentences into different RC categories. We designed a straightforward algorithm to do this task. The classification results are then manually checked.

We first identify all sentences annotated with `acl:relcl`, totaling 2036 instances. Subsequently, these sentences were categorized based on the Enhanced Universal Dependency (EUD) relations attributed to the relativized head noun. Our six target subtypes are derived from the EUD relation and whether it is a full or reduced RC: `nsubj` (full), `obj` (full and reduced), `obl` (full and reduced), `nsubj:pass` (full). All other variations, such as possessives, were consolidated under the *Others* category as shown in Table 2. Please note that the total count in the table does not match 2036 due to sentences that contain multiple types of RCs.

Reduced RC classification Enhanced UD relations were present for full RCs (having been added based on the relativizer's dependency relation in the basic layer) but were missing for reduced RCs. To infer the enhanced relation in reduced RCs, we implement rules to identify the locally missing (gapped) function of the RC. For example, in *He is the person **you like*** __, in the basic UD tree the verb *like* has a subject dependent but no object, which is used to infer that it is a reduced object RC.

Our implementation takes into account the overall transitivity of the RC predicate verb (whether it tends to be transitive or intransitive). We combine data from a verb transitivity file[4] and the dependency relations of verbs found in EWT. Treebank information is given precedence; if relations like `xcomp` or `ccomp` are among the top three most frequent associations with a verb, we classify it as transitive. Otherwise, we rely on the transitivity data from our table.

Next, we extract the set of relation labels of dependents of the RC predicate, applying recursion for instances of `xcomp` and `ccomp` so as to handle sentences such as *After I have done all the work **I promised to do**, I will take a break.* We then look for a missing relation: For transitive or ditransitive verbs, we categorize the clause as a reduced object RC if the verb has no `obj` dependent, and as a reduced oblique RC otherwise. Clauses associated with intransitive verbs are invariably considered oblique RCs. The procedure produces 340 reduced object RCs and 218 reduced oblique RCs.[5] The detailed statistics can be found in Table 2.

RC Category	Count	%
Subject RC	725	35.3
Object RC	161	7.8
Oblique RC	139	6.8
Passive RC	100	4.9
Reduced object RC	340	16.5
Reduced oblique RC	218	10.6
Others	373	18.1
Total	2056	100.0

Table 2: Distribution of RC types in the EWT data we used for evaluation.

Our method relies heavily on the information about the transitivity of verbs. Each verb type is assumed to be either transitive or intransitive, which makes ambitransitive verbs a tricky case. For example, in the two NPs *the day **he returned*** and *the piece **he returned***, the first relativizes an adverbial adjunct, while the second one is an object relative. However, in our verb transitivity table, *return* is a transitive verb, so the first example is mistakenly tagged as a reduced object RC. Most of the classification errors are caused by this problem.

Another tricky case is embedded complement clauses or control/raising constructions that are marked with `ccomp` or `xcomp` in UD separately. Consider the following two sentences:

(1) I will do all the work **I need to do** __

(2) I will talk to all the people **I need** __ **to do the work**.

If we extract all the dependencies of the predicate verb *need*, we will get the same relations: `nsubj`, `xcomp`, `obj`. However, as we can see, the missing object is in different embedded structures and therefore, the enhanced UD relation will be wrong in terms of the head. We therefore collected all RCs with `xcomp`/`ccomp` for manual correction.

We manually checked and corrected all examples in each reduced RC type. The results demonstrate high accuracy in discriminating the two classes, with a recall of 94% for reduced object RCs and 95% for reduced oblique RCs.

[4] `https://github.com/wilcoxeg/verb_transitivity` The CSV file contains the percentage of the time the verb is transitive, intransitive, and ditransitive in the Google syntactic ngrams corpus.

[5] Note that reduced subject RCs only occur in doubly embedded clauses (e.g. *the rooster **I thought was a hen***). These are rare and were dealt with manually.

AMR 3.0	All Sentences		RC Sentences		
Models	F (Full graph)	F (All reentrancies)	F (Full graph)	F (All reentrancies)	Subj RC Recall
AM-Parser§	74.9	57.0	73.5	57.7	65.2
amrlib-T5	82.0	71.4	77.6	70.4	71.0
amrlib-BART	82.3	73.5	80.6	73.3	**79.0**
Spring	83.0	68.0	72.5	65.5	65.2
AMRBART§	**84.2**	**74.3**	**80.8**	**73.4**	75.4

Table 3: Smatch F_1 scores and subject RC reentrancy recall of the models on AMR 3.0 test split. Two kinds of F_1 scores are shown: overall Smatch score comparing the full graph to the gold standard AMR, and the Reentrancies subscore (Damonte et al., 2017). These are shown for the full test set as well as the subset of test sentences containing a relative clause. The last column shows recall of reentrancies on subject relative clauses (138 examples in total; other RC subtypes were less frequent). "§" superscript means "structure-aware". The first four measures do not require token-level alignments between the graph and the text.

3.3. Models

In our experiments, we test five different models. The first, AM-Parser, derives a parse compositionally after predicting supertags and dependencies. The other four are sequence-to-sequence models, one of which has a structure-aware component in its training loss.

Structure-aware models AM-Parser (Groschwitz et al., 2018) is a neuro-symbolic compositional semantic parser that learns the sub-graphs of meaningful tokens and then combines them for a complete AMR. It is trained on two objectives: (a) learning the supertags aligned with each token; and (b) learning the dependency trees that connect the supertags to build a complete AMR graph. The supertagger and dependency parser are both trained on bert-large-uncased model.

AMRBART (Bai et al., 2022) is a graph-pretrained model based on BART (Lewis et al., 2020). Unlike traditional text-only pretraining, AMRBART masks parts of AMR graphs—like nodes and edges—during pretraining. It introduces a unified pretraining framework that combines the original text with its AMR graph, ensuring the model learns both linguistic content and graph structure. For pretraining, it uses 20k silver-standard AMR graphs created by Spring (Spring et al., 2021), and then it is fine-tuned with gold AMR data. The fine-tuned model shows more robust performance on unseen data, highlighting its potential for complex language tasks that require deep understanding.

Structure-unaware models We examined three structure-unaware models. They are pretrained language models fine-tuned on linearized AMRs with necessary preprocessing.

Spring (Spring et al., 2021) fine-tunes BART-base with vocabulary expansion. To achieve better results, instead of using linearized PENMAN notation, they adopt graph linearization by replacing variables with special tokens <Rx> where x is a number. In this way, the constants and variables in AMRs can be distinguished. Despite the preprocessing steps, the model still takes the input as sequence of strings without distinguishing the structural information and hence we categorize Spring as a structure-unaware model.

Similarly, amrlib fine-tunes the pre-trained language models such as BART-large and T5 models to translate natural language to linearized AMR.[6]

3.4. Evaluation

Our evaluation assesses whether the relativized noun in a sentence is reentrant, with two incoming edges—one originating from the main clause's predicate verb and another from the predicate within the RC. Take the sentence in Figure 1 as an example. After normalizing all the inverse edges, our script identifies the RC from the acl:relcl edge going from *person* to *likes*. It identifies the associated AMR nodes, person and like-01, and checks whether (1) the person node receives two incoming edges, and (2) there is an edge from like-01 to person. If so, the reentrancy expected for the RC is scored as recovered by the parser.

This analysis requires alignments between tokens in the sentence and their semantic nodes in order to determine, given a relative clause predicate p and its head noun n, which AMR edge (if any) is the associated reentrancy of the form $p \to n$. For AM-Parser, which inherently requires node-token alignment, we extract these alignments directly from its predictions. For the other parsers under study, we utilize LEAMR (Blodgett and Schneider, 2021), a probabilistic, fine-grained aligner optimized for English AMR.

Our evaluation metric is the **recall** in counting instances where the head noun's aligned node receives edges from both the main and RC predicate nodes. This approach allows us to effectively gauge

[6] https://github.com/bjascob/amrlib/wiki/The-parse_xfm-model

Model	Subj RC	Obj RC	Pass RC	Obl RC	RedObj RC	RedObl RC	All
AM-Parser§	57.4 (416/725)	55.3 (89/161)	74.0 (74/100)	33.3 (46/138)	50.6 (172/340)	34.4 (75/218)	51.8
	83.4 (605/725)	84.4 (136/161)	84.0 (84/100)	78.2 (108/138)	86.5 (294/340)	70.6 (154/218)	82.1
amrlib-BART	67.7 (491/725)	64.0 (103/161)	**80.0 (80/100)**	**65.2 (90/138)**	**62.1 (211/340)**	45.0 (98/218)	**63.8**
	87.2 (632/725)	83.9 (135/161)	94.0 (94/100)	87.0 (120/138)	80.6 (274/340)	67.0 (146/218)	83.2
amrlib-T5	**68.0 (493/725)**	**67.1 (108/161)**	77.0 (77/100)	55.1 (76/138)	59.4 (202/340)	45.4 (99/218)	62.7
	85.9 (623/725)	85.7 (138/161)	97.0 (97/100)	81.9 (113/138)	80.0 (272/340)	67.4 (147/218)	82.6
Spring	63.6 (461/725)	58.4 (94/161)	79.0 (79/100)	57.2 (79/138)	52.4 (178/340)	38.1 (83/218)	57.9
	81.5 (591/725)	76.4 (123/161)	94.0 (94/100)	76.8 (106/138)	73.5 (250/340)	56.4 (123/218)	76.5
AMRBART§	65.7 (476/725)	62.1 (100/161)	**80.0 (80/100)**	**65.2 (90/138)**	58.8 (200/340)	46.8 (102/218)	62.3
	85.5 (620/725)	80.1 (129/161)	94.0 (94/100)	87.0 (120/138)	79.1 (269/340)	69.7 (152/218)	82.3
Average	64.5 (467/725)	61.2 (99/161)	78.0 (78/100)	55.2 (76/138)	56.6 (193/340)	41.9 (91/218)	59.1

Table 4: Results by parser and RC type on the EWT dataset. Structure-aware parsers are notated with §. White rows report recall of RC-triggered reentrancy edges. Gray rows report *attainability* rates subject to the predicted nodes and their token alignments; this is an upper bound of recall. 3 graphs produced by AMRBART cannot be aligned with LEAMR, so we remove them from the evaluation set. The best results in each column and condition are indicated in **bold**.

Model	Subj RC	Obj RC	Passive RC	RedObj RC	All
AM-Parser§	96.0 (335/349)	96.0 (332/346)	97.1 (340/350)	92.4 (280/303)	95.5 (1,287/1,348)
amrlib-BART	**98.6 (344/349)**	98.0 (339/346)	**99.1 (347/350)**	**98.3 (298/303)**	**98.5 (1,328/1,348**)
amrlib-T5	98.3 (343/349)	97.7 (338/346)	98.9 (346/350)	94.0 (284/303)	97.3 (1,311/1,348)
Spring	97.7 (341/349)	**98.3 (340/346)**	**99.1 (347/350)**	98.0 (297/303)	98.3 (1,325/1,348)
AMRBART§	97.4 (340/349)	96.8 (335/346)	98.9 (346/350)	97.4 (295/303)	97.6 (1,316/1,348)
Average	97.6 (341/349)	97.3 (338/346)	98.6 (345/350)	96.0 (291/303)	98.0 (1,321/1,348)

Table 5: Recall by parser and RC type on the CRC dataset of synthetic sentences.

the parsers' proficiency in handling reentrancies within the constraints of available data.[7]

4. Results & Discussion

4.1. Overall Results

The initial assessment of the models was conducted on the AMR 3.0 test split (after running a dependency parser to find RCs), with outcomes presented in Table 3. The findings indicate that overall seq2seq models show better performance than the compositional AM-Parser model. Across metrics, AMRBART and amrlib-BART show good performance relative to other models.

It is also noteworthy that in the parsing of sentences with RCs, all models exhibit a decline in F-score, with Spring experiencing a sizable drop (from 83.0 to 72.5). This decrease may be attributed to the long-distance dependencies and more complex syntactic structures that relative clauses introduce.

The accuracy of 5 different models in processing various RC types in the new datasets is systematically examined and reported in Tables 4 and 5 for corpora with gold syntax annotations. If the predicate token or head token is not aligned to a node, it is impossible to get the reentrancy. Therefore, we also report the **attainability rate**, the rate at which node-token alignments could be recovered for both the RC head and predicate tokens, as seen in the gray rows of Table 4. If an RC reentrancy is *un*attainable, it means either that one or both of its tokens lack a corresponding node in the predicted AMR (usually a parser error), or that it was present but could not be aligned in post-processing (for systems where this step was necessary, namely the seq2seq models).

[7] We do not evaluate the role label on the reentrancy edge, because the role numbers in AMR predicates (mostly sourced from PropBank; Kingsbury and Palmer, 2002; Pradhan et al., 2022) are semantic rather than syntactic, and thus will not line up perfectly with the syntactic RC categories. However, the numbering conventions are weakly connected to syntactic functions: we expect that ARG0 should imply a subject RC; a subject RC should imply ARG0 or ARG1; a passive RC should usually imply ARG1; an object RC should imply ARG1 or ARG2; and ARG3, ARG4, etc. should generally imply an oblique RC (full or reduced). Non-core roles would likely correspond to obliques as well.

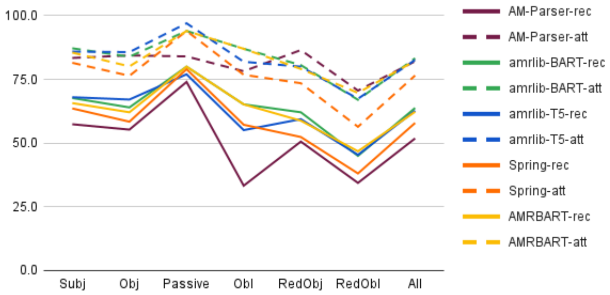


Figure 2: RC reentrancy recall (solid lines) and attainability rate (dashed) of all parsers, by RC subtype and overall.

EWT Overall, as reported in Table 4 and visualized in Figure 2, detecting the edges between the relative predicate and the head noun is challenging for all models, with recall below 64%.[8] This suggests that relative clause structures are especially difficult.

Comparison of parsers. Our results reveal that seq2seq parsers, whether they are structure-aware or not, outperform the compositional `AM-Parser`. Moreover, the overall performance of all seq2seq models is very similar. The performance of `AM-Parser` in parsing RCs appears less advantageous, which we conjecture may stem from the pretrained language model (i.e. `bert-large-uncased`) used. As we can see, the two `BART`-based parsers perform the best. Further exploration of the role of pretrained language models is left to future work.

Attainability. According to Table 4, we can see that even when both the head token and predicate token have predicted nodes, there remains considerable scope for further improvement given that UD parsing has reached over 95% in LAS since 2018 (e.g., Clark et al., 2018). This means that structural information is not fully captured by all models.

However, we recognize that the low recall might stem from the alignment model utilized. The attainability rate for oblique reduced RCs is particularly low, which likely affects recall scores. Misalignments between some subgraphs and tokens are observed; since our analysis targets subgraphs aligning with both the head and predicate tokens, such misalignments can diminish the scores. Additionally, it is possible that tokens are classified as edges rather than nodes, as illustrated in Figure 3 where no node but just an edge is aligned with the token *time*.

RC subtypes. Oblique, reduced oblique, and reduced object RCs are particularly hard. Psycholinguistic research has shown that oblique relative clauses are more challenging for humans to process due to the greater distance between the filler and the gap, compared to other types of relative clauses (e.g., Diessel and Tomasello, 2005); this distance may also be challenging for the AMR parsers. That reduced RCs are harder to parse than full RCs is likely due to the lack of explicit syntactic cues. It is interesting to see that passive RCs are easiest to parse of the RC categories. This is probably because both the relative pronoun and the passive construction provide more linguistic cues than other types of RCs. Subject RCs, the most frequent category in both the EWT and AMR 3.0 datasets (especially if the passive subjects are included), are easier than non-subject RCs. Psycholinguistic studies have shown subject RCs to be easier for humans to comprehend and acquire (Gordon and Lowder, 2012; Diessel and Tomasello, 2005), and Reali and Christiansen (2007) found that more frequent RC types are easier to process (but did not consider passive subject RCs).

```
(p0 / serve-01
    :ARG0 (p1 / person
              :wiki "George W. Bush"
              :name (p2 / name
                        :op1 "Bush"))
    :duration (p3 / nearly
                  :op1 (p4 / temporal-quantity
                           :quant 2
                           :unit (p5 / year)))
    :time (p6 / over-01
              :ARG1 (p7 / it)))
```

Figure 3: Predicted AMR for the sentence *By the time it was over, Bush had served nearly two years.*

CRC As for the synthetic data, scores are quite high across parsers and RC categories. `amrlib-BART` marginally outperforms other models on average. For object-reduced RCs, `AM-Parser` and `amrlib-T5` are notably weaker than the other systems. The CRC dataset does not contain any oblique RCs, so there is no relevant result on this category. The results for parsing different types of RCs presented in Table 5 align closely with those reported in Table 4.

4.2. Exploring Parsing Performance Variations in RCs

The models vary in absolute scores, but they follow a general trend: reentrancies in passive RCs are more often recovered than those in subject RCs, followed by object RCs and oblique RCs. Reduced RCs are harder to predict. We observe a similar pattern in the CRC data both in dependency and semantic parsing. Next we explore two possible factors influencing parsing performance across RCs, namely, dependency distance and training data distribution.

[8] For the `amrlib-BART` model (overall recall of 63.8%), we also computed recall of AMR edges for `ccomp` complement clauses, which was much higher: 77.4% (1445/1868), with an attainability rate of 82.7 (1545/1868).

RC Category	Dep Dist	Mean Recall
Reduced oblique RC	3.06	41.9
Reduced object RC	3.13	56.6
Subject RC	4.30	64.5
Passive RC	5.78	78.0
Object RC	5.21	61.2
Oblique RC	6.98	55.2

Table 6: Mean dependency distance of 6 types of RCs in our experiments

RC Category	Count
Subject RC	4,226
Object RC	516
Oblique RC	729
Passive RC	534
Reduced object RC	1,371
Reduced oblique RC	1,092

Table 7: Distribution of 6 RC types in AMR 3.0 train split

Dependency distance Dependency distance refers to the linear distance between two words connected by a dependency relation, which functions as an important indicator of syntactic difficulty (Liu et al., 2017). Existing research has reported that longer dependency distance makes subject RCs easier to process than object RCs in English (Gibson, 1998) and vice versa in Chinese (Hsiao and Gibson, 2003). In this paper, we calculate the mean dependency distance between the predicate in the RC and the head noun in the matrix clause in each type of RC. It is surprising that the reduced RCs have the shortest dependency distance even if we assume the existence of the relative pronoun (i.e., we add 1 to the existing dependency distance). The shorter distance might justify dependency distance minimalization (Temperley, 2007) because the omission of the relative pronoun makes the sentence harder to process and therefore only shorter dependency distance makes them easier to process.

Regarding the full RCs, as shown in Table 6, in the EWT dataset, the dependency distance largely meets the observation made by previous research that subject RC is easier than object RC. Notably, passive RCs, despite their longer dependency distances, exhibit high parsing accuracy. This could be attributed to passive RCs essentially acting as subject RCs, with the relative pronoun serving as the subject. When considering subject and passive RCs together, the average dependency distance decreases to 4.46, making these types the most straightforward for parsers.

Training data distribution We investigated the distribution of different RC types within the AMR 3.0 training split. Given the absence of gold-standard dependency annotations in AMR 3.0, we obtained automatic dependency trees using `Stanza`.[9] For full RCs, classification was based on the dependency relationship between the relative pronoun and its predicate. The identification of reduced RCs employed the methodology outlined in §3.2. As Table 7 illustrates, the prevalence of RC types in AMR 3.0 closely mirrors that of EWT, with subject RCs being the most common.

It is intriguing that despite being more common, subject RCs are still tougher to handle than their passive forms. This revelation suggests that the frequency of a structure does not necessarily make it easier to process, hinting at deeper complexities in understanding syntactic patterns.

5. Conclusion

In our study, we compared two structure-aware AMR parsers (`AM-Parser` and `AMRBART`) and typical structure-unaware seq2seq models (`Spring`, `amrlib-BART`, and `amrlib-T5`) in parsing relative clauses. We find that relative clauses are challenging for current parsers. Seq2seq models, on the whole, outperform the compositional model. Interestingly, there is little difference in performance between seq2seq models that are aware of structure and those that are not. Furthermore, our analysis reveals that (reduced or full) oblique and reduced object RCs are the most challenging RC types. Examining the relationship to dependency length, we find that the full RCs with shorter dependency distances are easier to parse; however, reduced RCs with the shortest dependency distance are more challenging for all parsers. As part of our study, we have produced gold EUD annotations for reduced RCs in the English Web Treebank; these will be released upon publication.

Future work might expand the scope of inquiry to more diverse reentrancy types by leveraging the (E)UD annotations. It would also be interesting to see if adding (E)UD information to AMR parsing helps the structure-unaware parsers to learn the complex structural information (cf. Findlay and Haug, 2021).

Acknowledgments

We thank Jonas Groschwitz and Austin Blodgett for their invaluable technical assistance, and members of the NERT lab for their insightful feedback. We are also grateful to the anonymous reviewers whose suggestions have significantly enhanced this paper. This research was supported in part by NSF award IIS-2144881.

[9] `stanza-1.6.0`: https://github.com/stanfordnlp/stanza/releases/tag/v1.6.0

Bibliographical References

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)*, pages 6001–6015, Dublin, Ireland. 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 *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria. Association for Computational Linguistics.

Ann Bies, Justin Mott, Colin Warner, and Seth Kulick. 2012. English Web Treebank. LDC2012T13.

Austin Blodgett and Nathan Schneider. 2021. Probabilistic, structure-aware algorithms for improved variety, accuracy, and coverage of AMR alignments. In *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)*, pages 3310–3321, Online. Association for Computational Linguistics.

Shu Cai and Kevin Knight. 2013. 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, Sofia, Bulgaria. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Christopher D. Manning, and Quoc Le. 2018. Semi-supervised sequence modeling with cross-view training. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1914–1925, Brussels, Belgium. Association for Computational Linguistics.

Marco Damonte, Shay B. Cohen, and Giorgio Satta. 2017. An incremental parser for Abstract Meaning Representation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 536–546, Valencia, Spain. Association for Computational Linguistics.

Forrest Davis and Marten van Schijndel. 2020. Recurrent neural network language models always learn English-like relative clause attachment. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1979–1990, Online. Association for Computational Linguistics.

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

Holger Diessel and Michael Tomasello. 2005. A new look at the acquisition of relative clauses. *Language*, pages 882–906.

Jamie Y. Findlay and Dag T. T. Haug. 2021. How useful are enhanced Universal Dependencies for semantic interpretation? In *Proceedings of the Sixth International Conference on Dependency Linguistics (Depling, SyntaxFest 2021)*, pages 22–34, Sofia, Bulgaria. Association for Computational Linguistics.

Edward Gibson. 1998. Linguistic complexity: Locality of syntactic dependencies. *Cognition*, 68(1):1–76.

Peter C Gordon and Matthew W Lowder. 2012. Complex sentence processing: A review of theoretical perspectives on the comprehension of relative clauses. *Language and Linguistics Compass*, 6(7):403–415.

Jonas Groschwitz, Shay Cohen, Lucia Donatelli, and Meaghan Fowlie. 2023. AMR parsing is far from solved: GrAPES, the granular AMR parsing evaluation suite. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10728–10752, Singapore. Association for Computational Linguistics.

Jonas Groschwitz, Matthias Lindemann, Meaghan Fowlie, Mark Johnson, and Alexander Koller. 2018. AMR dependency parsing with a typed semantic algebra. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1831–1841, Melbourne, Australia. Association for Computational Linguistics.

Franny Hsiao and Edward Gibson. 2003. Processing relative clauses in Chinese. *Cognition*, 90(1):3–27.

Edward L. Keenan and Bernard Comrie. 1977. Noun phrase accessibility and universal grammar. *Linguistic Inquiry*, 8(1):63–99.

Najoung Kim, Roma Patel, Adam Poliak, Patrick Xia, Alex Wang, Tom McCoy, Ian Tenney, Alexis Ross, Tal Linzen, Benjamin Van Durme, Samuel R. Bowman, and Ellie Pavlick. 2019. Probing what different NLP tasks teach machines about function word comprehension. In *Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (*SEM 2019)*, pages 235–249, Minneapolis, Minnesota. Association for Computational Linguistics.

Paul Kingsbury and Martha Palmer. 2002. From TreeBank to PropBank. In *Proc. of LREC*, pages 1989–1993, Las Palmas, Canary Islands.

Kevin Knight, Bianca Badarau, Laura Baranescu, Claire Bonial, Madalina Bardocz, Kira Griffitt, Ulf Hermjakob, Daniel Marcu, Martha Palmer, Tim O'Gorman, and Nathan Schneider. 2021. Abstract Meaning Representation (AMR) Annotation Release 3.0. Linguistic Data Consortium, LDC2020T02.

Young-Suk Lee, Ramón Astudillo, Hoang Thanh Lam, Tahira Naseem, Radu Florian, and Salim Roukos. 2022. Maximum Bayes Smatch ensemble distillation for AMR parsing. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5379–5392, Seattle, United States. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Bingzhi Li, Lucia Donatelli, Alexander Koller, Tal Linzen, Yuekun Yao, and Najoung Kim. 2023. SLOG: A structural generalization benchmark for semantic parsing. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3213–3232, Singapore. Association for Computational Linguistics.

Haitao Liu, Chunshan Xu, and Junying Liang. 2017. Dependency distance: A new perspective on syntactic patterns in natural languages. *Physics of life reviews*, 21:171–193.

Marius Mosbach, Stefania Degaetano-Ortlieb, Marie-Pauline Krielke, Badr M. Abdullah, and Dietrich Klakow. 2020. A closer look at linguistic knowledge in masked language models: The case of relative clauses in American English. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 771–787, Barcelona, Spain (Online). International Committee on Computational Linguistics.

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 *Proc. of LREC*, pages 4027–4036, Marseille, France.

Sameer Pradhan, Julia Bonn, Skatje Myers, Kathryn Conger, Tim O'Gorman, James Gung, Kristin Wright-Bettner, and Martha Palmer. 2022. PropBank comes of age—larger, smarter, and more diverse. In *Proc. of *SEM*, pages 278–288, Seattle, Washington.

Grusha Prasad, Marten van Schijndel, and Tal Linzen. 2019. Using priming to uncover the organization of syntactic representations in neural language models. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 66–76, Hong Kong, China. Association for Computational Linguistics.

Shauli Ravfogel, Grusha Prasad, Tal Linzen, and Yoav Goldberg. 2021. Counterfactual interventions reveal the causal effect of relative clause representations on agreement prediction. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 194–209, Online. Association for Computational Linguistics.

Florencia Reali and Morten H Christiansen. 2007. Processing of relative clauses is made easier by frequency of occurrence. *Journal of memory and language*, 57(1):1–23.

Peter Shaw, Ming-Wei Chang, Panupong Pasupat, and Kristina Toutanova. 2021. Compositional generalization and natural language variation: Can a semantic parsing approach handle both? In *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)*, pages 922–938, Online. Association for Computational Linguistics.

Natalia Silveira, Timothy Dozat, Marie-Catherine de Marneffe, Samuel R. Bowman, Miriam Connor, John Bauer, and Christopher D. Manning. 2014. A gold standard dependency corpus for English. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, pages 2897–2904, Reykjavík, Iceland.

Nicolas Spring, Annette Rios, and Sarah Ebling. 2021. Exploring German multi-level text simplification. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2021)*, pages 1339–1349, Held Online. INCOMA Ltd.

Ida Szubert, Marco Damonte, Shay B. Cohen, and Mark Steedman. 2020. The role of reentrancies in Abstract Meaning Representation parsing. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2198–2207, Online. Association for Computational Linguistics.

David Temperley. 2007. Minimization of dependency length in written English. *Cognition*, 105(2):300–333.

Alex Warstadt and Samuel R Bowman. 2019. Linguistic analysis of pretrained sentence encoders with acceptability judgments. *arXiv preprint arXiv:1901.03438*.

Yuekun Yao and Alexander Koller. 2022. Structural generalization is hard for sequence-to-sequence models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5048–5062, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Gaining More Insight into Neural Semantic Parsing with Challenging Benchmarks

Xiao Zhang, Chunliu Wang, Rik van Noord, Johan Bos
Center for Language and Cognition, University of Groningen
{xiao.zhang, chunliu.wang, r.i.k.van.noord, johan.bos}@rug.nl

Abstract

The Parallel Meaning Bank (PMB) serves as a corpus for semantic processing with a focus on semantic parsing and text generation. Currently, we witness an excellent performance of neural parsers and generators on the PMB. This might suggest that such semantic processing tasks have by and large been solved. We argue that this is not the case and that performance scores from the past on the PMB are inflated by non-optimal data splits and test sets that are too easy. In response, we introduce several changes. First, instead of the prior random split, we propose a more systematic splitting approach to improve the reliability of the standard test data. Second, except for the standard test set, we also propose two challenge sets: one with longer texts including discourse structure, and one that addresses compositional generalization. We evaluate five neural models for semantic parsing and meaning-to-text generation. Our results show that model performance declines (in some cases dramatically) on the challenge sets, revealing the limitations of neural models when confronting such challenges.

Keywords: Annotated Corpus, Discourse Representation Theory, Semantic Parsing, Text Generation

1. Introduction

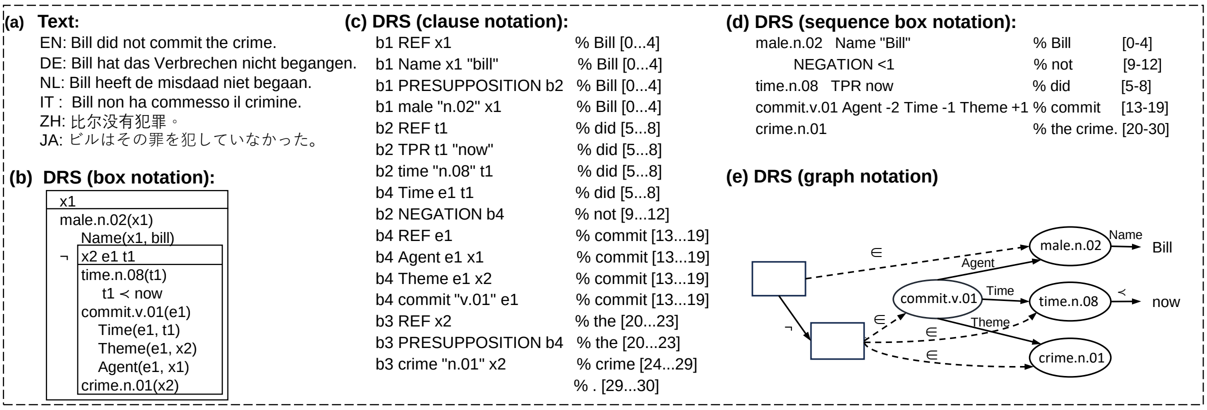
The Parallel Meaning Bank (PMB, Abzianidze et al., 2017) is a semantically annotated parallel corpus for multiple languages. It consists of a large collection of parallel texts, each accompanied by a formal meaning representation based on a variation of Discourse Representation Theory (DRT, Kamp and Reyle, 1993), called Discourse Representation Structure (DRS). It can be used for corpus-based studies on formal semantic phenomena, or to develop and evaluate semantic processing tasks such as text-to-meaning parsing and meaning-to-text generation. As a matter of fact, the PMB has been widely used in semantic parsing (Abzianidze et al., 2019; van Noord, 2019; van Noord et al., 2020; Wang et al., 2021b; Poelman et al., 2022), natural language generation (Wang et al., 2021a, 2023), and semantic tagging (Bjerva et al., 2016; Abzianidze and Bos, 2017; Abdou et al., 2018; Huo and de Melo, 2020).

The rapid development of neural models and their incredible performance seem to make the impression that tasks like semantic parsing are practically solved. For instance, the state-of-the-art DRS parser (Wang et al., 2023) achieves a remarkable score of approximately 95.0 on the English test set of the PMB and manual analysis reveals that the parser made very few errors except for words outside the vocabulary. Are neural models mastering semantic parsing (and indeed natural language generation), even for complex formal meaning representations like those present in the PMB? Or is there something else going on, and does this perception not align with the actual state of affairs?

We carried out a critical examination of the PMB and revealed three (related) problems: (1) there is a “data leakage” from the training data to the development and test splits; (2) the random splits of the data lead to a non-optimal division; and (3) the test set is often regarded as “easy” as it contains a large amount of relatively short sentences. Let us elaborate on this a bit.

In the current release of the PMB, the data splits were randomly decided and considered "standard". However, this random split may result in overlap and imprecise error estimates (Søgaard et al., 2021) and and cannot adequately represent the distribution of the dataset. For instance, the sentence “*I like chocolate ice cream!*” is allocated to the training set, while the very similar sentence “*I like chocolate ice cream.*” is assigned to the test set. Equally alarmingly, some instances in the development and test sets mirror those in the training set, potentially skewing parser evaluations. Consequently, this may lead to parser evaluation results that are overly optimistic. We completely agree with Opitz and Frank (2022) and Groschwitz et al. (2023), who both argue that "AMR Parsing is far from solved" hits the nail on the head, and even goes beyond Abstract Meaning Representation (AMR) and also applies to DRS. We think the current PMB test set lacks difficulty, because it puts emphasis on brief and simplistic sentences with an average length of less than ten words. The reason for this is that all instances of the test set have the “gold” annotation status, obtained via intensive manual correction, and the longer a document the harder it is to get an error-free annotation for it.

The aim of this paper is (a) to show that the



(a) Text:

EN: Bill did not commit the crime.
DE: Bill hat das Verbrechen nicht begangen.
NL: Bill heeft de misdaad niet begaan.
IT : Bill non ha commesso il crimine.
ZH: 比尔没有犯罪。
JA: ビルはその罪を犯していなかった。

(b) DRS (box notation):

(c) DRS (clause notation):

```
b1 REF x1                  % Bill [0...4]
b1 Name x1 "bill"          % Bill [0...4]
b1 PRESUPPOSITION b2       % Bill [0...4]
b1 male "n.02" x1          % Bill [0...4]
b2 REF t1                  % did [5...8]
b2 TPR t1 "now"            % did [5...8]
b2 time "n.08" t1          % did [5...8]
b4 Time e1 t1              % did [5...8]
b2 NEGATION b4             % not [9...12]
b4 REF e1                  % commit [13...19]
b4 Agent e1 x1             % commit [13...19]
b4 Theme e1 x2             % commit [13...19]
b4 commit "v.01" e1        % commit [13...19]
b3 REF x2                  % the [20...23]
b3 PRESUPPOSITION b4       % the [20...23]
b3 crime "n.01" x2         % crime [24...29]
                           % . [29...30]
```

(d) DRS (sequence box notation):

```
male.n.02   Name "Bill"                    % Bill         [0-4]
      NEGATION <1                          % not          [9-12]
time.n.08   TPR now                        % did          [5-8]
commit.v.01 Agent -2 Time -1 Theme +1      % commit       [13-19]
crime.n.01                                 % the crime.   [20-30]
```

(e) DRS (graph notation)

Figure 1: (a) An example sentence "*Bill did not commit the crime.*" taken from the PMB in six languages with its DRS in (b) box notation, (c) clause notation, (d) sequence box notation, and (e) graph notation.

random split indeed leads to an undesired simplification of the task, and (b) to demonstrate that the task of semantic parsing is far from being solved by providing a new challenging test set.

Inspired by the work of Søgaard et al. (2021), we design three new test sets: one standard test set and two challenge sets. The former is implemented by a two-round sorting approach to establish a more systematic split, ensuring the reliability and independence of standard development and test sets. The latter comprises a test set with substantially *longer texts* and a test set based on *compositional recombination*. The long-text set is derived by choosing documents with long texts from the PMB and manually correct the automatically assigned meaning representation. This set aims to assess the parser's performance on long and multi-sentence texts. The compositional set consists of texts formed by recombining the Combinatory Categorical Grammar (CCG, Steedman, 1996) derivation tree that is provided with the PMB data. This kind of tree recombination technique has been empirically validated for semantic data augmentation by Juvekar et al. (2023). Differently, we employ this technology for the creation of test sets, with the intent of assessing the semantic parser model's capability in compositional generalization (Furrer et al., 2020). To our knowledge, we are the first to utilize CCG to create data for compositional generalization testing. By empirical analysis of the performance of neural semantic parsers and generators based on five different language models, we show the effect of our newly created systematic split and challenge sets.

2. Background and Related Work

In this section, we first provide an overview of DRS, PMB, and CCG, review the works in parsing and generation, and introduce different data split methods. Subsequently, we introduce existing tasks and corpora related to long text semantic and compositional generalization.

2.1. Discourse Representation Structure

DRS is the formal meaning representation in the PMB, capturing the essence of the text and covering linguistic phenomena like anaphors and temporal expressions. Unlike many other formalisms such as Abstract Meaning Representation (AMR, Banarescu et al., 2013) used for large-scale semantic annotation efforts, DRS covers logical negation, quantification, and discourse relations, has complete word sense disambiguation, and offers a language-neutral meaning representation.

DRS can be represented in multiple formats as is shown in Figure1. In the box notation, DRS uses boxes containing discourse referents and conditions. Discourse referents, like *x1*, serve as markers for entities introduced in the discourse. Conditions convey information over the referents: to what concepts they belong and what relations they have to other referents, expressed by roles or comparison operators. Concepts are grounded by WordNet synsets, such as *male.n.02*. Thematic roles are derived from VerbNet (Bonial et al., 2011), for instance *Agent*. Operators, like $\prec$, $=$, $\neq$ and $\sim$, are utilized to formulate comparisons among entities. Furthermore, conditions can also be complex, serving to represent logical (negation, $\neg$) or rhetorical relations among different sets of conditions.

The clause notation is converted from box notation to adapt to machine learning models (van Noord et al., 2018). In the conversion, the label of the box, wherein the discourse referents and conditions are located, is positioned to precede them.

To simplify DRS, Bos (2023) introduced a variable-free DRS format called Sequence Box Notation (SBN), where the sequencing of terms is important. The meaning of each word adheres to an entity-role-index structure, with indices connecting entities and roles decorating connection.

The discourse relations (such as NEGATION and ELABORATION) are slightly different, indicating the beginning of a new context. The subsequent indices, marked with comparison symbols ($<$,$>$), link the newly established context to another context. SBN can also be interpreted as a directed acyclic graph, as depicted in Figure 1(e).

2.2. Combinatory Categorical Grammar

CCG is a lexicalised grammar formalism (Steedman, 1996) used in the PMB to steer the compositional semantics. It comprises just a few basic categories — N (noun), NP (noun phrase), PP (prepositional phrases) and S (sentence) — from which function categories can be composed using the backward slash for combining with phrases to the left and the forward slash for combining with phrases to its right. For instance, a typical determiner gets the lexical category NP/N to look for a noun (N) on its right resulting in a noun phrase (NP). CCG expressions can be combined with each other obeying the combinatorial rules, of which there are just a handful. The most common rules are forward and backward application:

$$\text{Forward App.} \quad (>): \quad (X/Y)\ Y \Rightarrow X \qquad (1)$$

$$\text{Backward App.} \quad (<): \quad Y\ (X\backslash Y) \Rightarrow X \qquad (2)$$

In the PMB, each CCG category is paired with a meaning representation with a semantic type that mirrors the internal structure of the category. This makes it a formidable linguistic formalism to implement compositional semantics.

2.3. The Parallel Meaning Bank

The PMB has evolved through four versions. Originating from the English-specific Groningen Meaning Bank (GMB, Basile et al., 2012), the PMB expanded it by embracing multiple languages. The initial version introduced German, Dutch, and Italian with their gold standard DRS in box format. The second version added silver and bronze standard data, which are partially corrected and uncorrected. Subsequent versions, namely the third and fourth versions, have witnessed an increased volume of manually annotated data and a shift from box to clause notation.

The PMB employs seven layers to process raw text, with each layer contributing an additional piece of syntactic/semantic information, building upon the results from the preceding layer (Abzianidze et al., 2020). The seven layers encompass tokenization, symbolization, word sense disambiguation, co-reference resolution, thematic role labeling, syntactic analysis and semantic tagging. Manual corrections are allowed at every layer. The final layer yields a CCG derivation tree, which is then utilized as input for the Boxer (Bos, 2015) and is converted into DRS. Initially tailored for English, PMB aligns it with other languages using an annotation projection method (Abzianidze et al., 2020).

In the field of semantic-related tasks, PMB has been widely used. However, it is not without limitations. Haug et al. (2023) emphasizes that a large portion of PMB data consists of short sentences, which compromises its ability to accurately represent real-world data.

2.4. Parsing and Generation with DRS

Semantic parsing with DRS initially employed rule-based parsers, such as Boxer (Bos, 2008). With the advent of neural models, the focus shifted to seq2seq approaches using LSTMs (van Noord et al., 2019, 2020). However, recent innovations include tree-based (Liu et al., 2018, 2019; Poelman et al., 2022) and graph-based techniques (Fancellu et al., 2019; Fu et al., 2020). In the ongoing exploration of neural networks, parsers have increasingly embraced transformer-based models like T5 (Raffel et al., 2019), BART (Lewis et al., 2020), and their variants. A significant breakthrough was DRS-MLM (Wang et al., 2023), a model that pre-trained mBART on PMB data and achieved state-of-the-art results in multiple languages. For meaning-to-text generation, Wang et al. (2021a) utilized a bi-LSTM on DRS's linearized format and found character-level decoders optimal. The mentioned DRS-MLM can also be used for DRS-to-text generation in pre-training steps outperforming other generators.

2.5. Data Split Methods

In most of the standardized datasets (Marcus et al., 1994; Fares et al., 2018), a consistent test set is typically maintained to enable comparisons between models (van der Goot, 2021). Traditionally, this kind of test set is created by random sampling (Elazar and Goldberg, 2018; Poerner et al., 2018), as is the current practice in the PMB. However, as we mentioned in the introduction, this random selection will lead to a data leakage from train to test. Multiple random split (Gorman and Bedrick, 2019) may be a fairer approach, but this will make comparison of models more difficult. To address these problems, Søgaard et al. (2021) advocates for the utilization of a biased or adversarial split besides the standard split, aiming to reduce the deviation between the test set and real-world data. We adopted this suggestion and developed an unbiased standard test set along with two biased challenge test sets, as detailed in Section 3.

2.6. Semantic Corpora with Long Texts

Few corpora focus on the semantics of long texts, primarily because of difficult annotations and constraints in meaning representation itself (For instance, AMR was initially designed for single sentences). O'Gorman et al. (2018) addressed this by manually annotating coreference, implicit roles, and bridging relations to create the multi-sentence AMR corpus. Other annotated corpora address discourse structure and rhetorical structure (Prasad et al., 2008), but ignore sentence semantics. As mentioned in Section 2.1, DRS is naturally designed for discourse, eliminating the need for additional annotation rules when annotating the meaning of long texts. Therefore, our annotation is more straightforward, as introduced in Section 3.

2.7. Compositional Generalization

Several studies have demonstrated that neural models tend to memorize patterns observed during training, struggling to generalize effectively to unfamiliar patterns (Lake and Baroni, 2018; Furrer et al., 2020). The combinationality in language significantly exacerbates this struggle. To assess this, tasks and datasets like the SCAN (Lake and Baroni, 2017) and the COGS (Kim and Linzen, 2020) have been developed. Kim and Linzen (2020) pointed out despite excellent standard test performances, their models reveal gaps in compositional generalization ability. This kind of gap led to our creation of the second challenge test set in Section 3 and experiments in Section 4.

3. Improving Semantic Evaluation

In this section we outline the methods to create better test sets. Besides the standard test set created with a different data split, we also show how we built additional challenge test sets. The resulting data set will be released as PMB 5.0.0[1].

3.1. Splitting Data Systematically

As mentioned in Section 1, the random split method employed by the PMB requires improvement. We have devised a strategy that reduces overlap between training and standard development/test sets, without introducing additional biases.

Our data split strategy involves two rounds of sorting. First, documents are sorted by character length. Afterward, the ordered collections are divided into groups of ten documents, which are then re-sorted based on their internal edit distances. The first sorting aims to maintain a consistent length distribution across the training, development, and test sets, while also ensuring some degree of uniformity in their semantic distribution. This is crucial to minimize bias introduced in the standard test data. The second sorting is particularly designed to create a certain degree of separation between the datasets, aiming at decreasing the word overlap. We allocate the first eight documents to the training set, and the remaining two are randomly distributed between the development and test sets. In Section 4, our experiments and analysis prove that the systematic split reduces the overlap between the training and development/test sets.

The distributions of gold data under the systematic split are shown in Table 1. For English, we adopt an 8:1:1 split ratio, while for the other three languages, we use a 4:3:3 ratio to ensure the test data is sufficient.

3.2. Creating Challenge Sets

We create two challenge sets for English: one focusing on long texts and another dedicated to compositional recombination by CCG.

3.2.1. Long-Text Challenge Set

Given that the gold data in the PMB predominantly consists of short sentences, with an average sentence length ranging between five and six words, it constrains our evaluation of the model's capability with long texts. In response, we select silver documents that notably exceed this average length for manual annotation, and change these into gold by correcting discourse structure, rhetorical relations, ellipsis, and inter-sentential pronouns (see Appendix A.2 for an example). Our long-text set includes 138 data samples with an average text length of 61 words, roughly ten times longer than the standard test set. The average lengths of train, development and test sets are shown in Table 1.

3.2.2. Compositional Challenge Set

As introduced in section 2, the final layer of the PMB produces the CCG derivation tree that is enriched with syntactic and semantic information, which is subsequently passed to the boxer to produce DRS. Therefore, recombining the gold CCG tree with other trees can yield distinct CCG trees, with associated text and DRS. In contrast to the creation of the long-text set, the quality of the DRS produced by this method closely approximates the gold standard, which greatly reduces the need for further manual annotation.

The original CCG derivation tree contains the compositional categories of words and phrases in a sentence, as shown in Figure 2 (a). We introduce two recombination operations: substitution and extension, shown in Figure 2 (c) and (d). In the

[1] The release is available at https://pmb.let.rug.nl/releases/

	Train	Dev	Standard Test	Long Test	Compositional Test
English (EN)	9,057 (5.64)	1,132 (5.38)	1,132 (5.15)	138 (60.78)	1,148 (6.48)
German (DE)	1,206 (5.06)	900 (4.79)	900 (4.87)	—	—
Dutch (NL)	586 (5.62)	435 (5.09)	435 (5.08)	—	—
Italian (IT)	745 (4.73)	555 (4.52)	555 (4.53)	—	—

Table 1: Distribution of train, development, and test sets in PMB 5.0.0 using the systematic split, together with two challenge sets. The average sentence length of each set are provided in brackets.

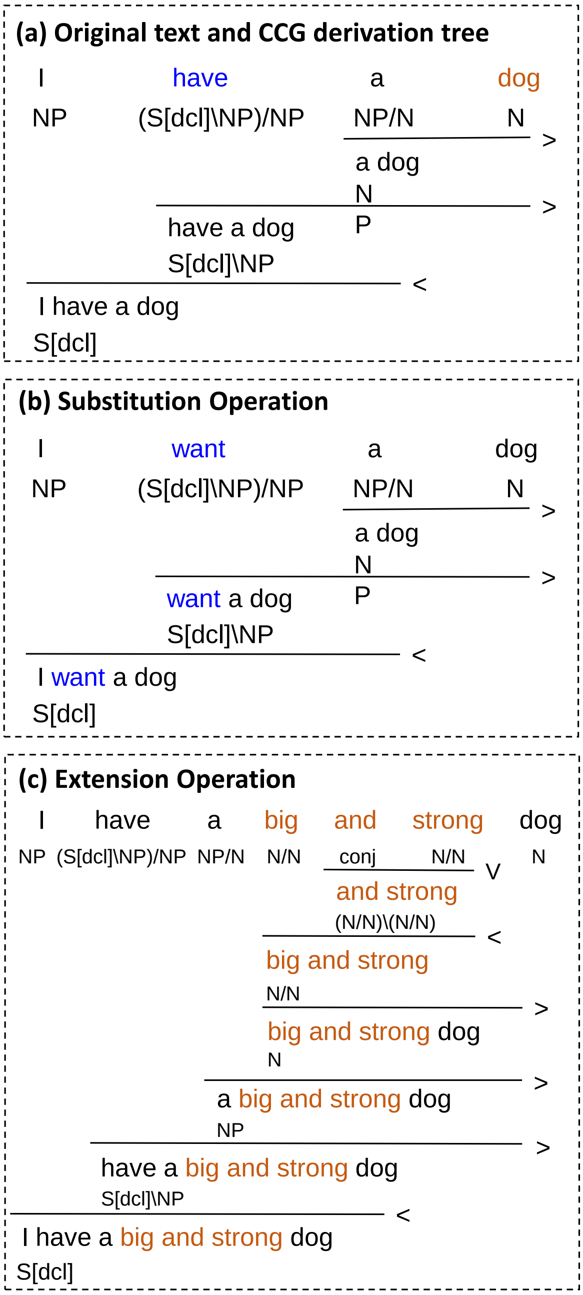


Figure 2: Two recombination operations performed on the CCG derivation tree of example sentence "*I have a dog*": (b) substitution (c) extension. We retained only the CCG categories and their corresponding words/phrases, excluding other semantic information.

substitution operation, the leaves or subtrees within a CCG derivation tree are replaced by counterparts from other different trees, provided they share the same CCG category. For instance, the word *have* swaps with *want*, as highlighted in blue. The extension operation takes a singular leaf from the tree and develops it into a larger subtree. As shown in Figure 2 (c), *dog* with the N category is extended to a subtree rooted at N, resulting in the phrase *big and strong dog*. The pseudo-code detailing these two operations is provided in Appendix A.1.

However, this method will generate many semantically abnormal sentences though they adhere strictly to syntactic structure. In this case, we use masked language models to estimate sentence pseudo-log-likelihood (PLL) scores (Salazar et al., 2020; Kauf and Ivanova, 2023). In practice, BERT (Devlin et al., 2018) is utilized as the scoring model, with a manually determined threshold. Specifically, the threshold is adjusted to eliminate 95% of the generated sentences, retaining only the top 5% that are highly deemed semantically correct.

Using this approach, we recombine the CCG trees of training samples and choose from the generated data, with the details presented in Table 1. Table 2 and 3 show some example texts produced through substitution and extension operations. Beyond individual operations, we also conduct multiple iterations on a sentence. The symbol $\times$ indicates the number of times an operation is applied to the same sentence.

4. Experiments and Analysis

This section offers an introduction to the selected seq2seq models, experimental settings, results and analysis for the text-to-DRS parsing and DRS-to-text generation.

4.1. Model Selection

The current approach to semantic parsing and text generation with DRS mainly involves fine-tuning a pre-trained language model. Our initial experiment employs a model based on BERT embeddings and LSTM architecture, following the methodology of van Noord et al. (2020). Then we utilize T5 and BART, two pre-trained transformer-based models. Specifically, we choose their multilingual variants:

Category	Operation	Training Set	Compositional Set
Noun	N⇒N	Bill was killed by an intruder.	Bill was killed by an Irishman.
Pronoun	NP⇒NP	My bag is very heavy.	His bag is very heavy.
Verb	(S\NP)/NP⇒(S\NP)/NP	The police are following us.	The police are visiting us.
Adjective	S\NP⇒S\NP	My tie is orange.	My tie is wet.
Adverb	(S\NP)/(S\NP)⇒(S\NP)/(S\NP)	The rent is very high.	The rent is extremely high.
Preposition	PP/NP⇒PP/NP	The boy bowed to me.	The boy bowed behind me.
Determiners	NP/N⇒NP/N	The answer is clear.	Neither answer is clear.
Modal	(S\NP)/(S\NP)⇒(S\NP)/(S\NP)	It will be scary.	It should be scary.
Substitution×2	N⇒N + (S\NP)/NP⇒(S\NP)/NP	Russia fears the system.	Cuba replaced the system.
Substitution×3	NP⇒NP + PP/NP⇒PP/NP + S\NP⇒S\NP	I took the elevator to the fourth floor.	They took another elevator to the last floor.

Table 2: Examples of substitution operations with CCG categories and operations. Note the table only shows the most common combinations for both two-fold (substitution × 2) and three-fold (substitution × 3) iterations. The color blue indicates the operation depicted in Figure 2 (b).

Category	Training Set	Compositional Set
Noun	My brother is rich.	My bad brother is rich. My brother who is speaking English is rich.
Verb	Coffee will be served after the meal.	Coffee will be secretly served after the meal. Coffee will be served by Elizabeth after the meal.
Adjective	Tom was thoughtful.	Tom was very thoughtful. Tom was thoughtful and innocent.
Extension×2	Tom is courteous.	Tom himself is more courteous. Tom who did it is courteous.
Extension×3	There are thirty names on the list.	There are about thirty new names on the short list. There are over thirty other names by Berlioz on the list.

Table 3: Examples of extension operations. We have excluded the operations of CCG categories due to the vast number of extension variations, which are nearly impossible to cover comprehensively. Instead, we present the most prevalent extension types for each category. The color orange indicates the operation depicted in Figure 2 (c).

mT5 (Xue et al., 2021), byT5 (Xue et al., 2022), mBART (Liu et al., 2020), and DRS-MLM (Wang et al., 2023) which is pre-trained on DRS data using the mBART architecture. In the case of DRS-MLM, for it is initially pre-trained on a train set under random split, we re-pre-train it using the train set based on our systematic split. To maintain consistent model sizes, we selected the large version across all models.

4.2. Evaluation Metrics

The evaluation process for Text-to-DRS parsing consists of two primary phases (Poelman et al., 2022). Firstly, the generated DRSs and gold standard DRSs are transformed into Penman notation (Kasper, 1989). Subsequently, we utilize SMATCH (Cai and Knight, 2013), an evaluation tool for AMR parsing, to calculate the match between the output and the gold standard by quantifying the overlap of triples. Evaluation of the generation task is conducted using BLEU (Papineni et al., 2002), METEOR (Lavie and Agarwal, 2007), and COMET(Rei et al., 2020).

4.3. Experiment Settings

We carried out three primary experiments. (1) We fine-tuned the selected language models for four languages: EN, DE, NL, and IT, and evaluated them using the standard test set. Following the training configurations set by van Noord et al. (2018); Poelman et al. (2022); Wang et al. (2023), we trained the models on gold and silver data for EN, and trained on gold, silver, and bronze data for DE, NL, and IT. This was subsequently followed by a fine-tuning phase exclusively on gold data; (2) We calculated and compared the word overlap rate of the train sets and test sets under systematic and random split. Then, we showed the performance of the two top-performing models from the first experiments under these two splits. To ensure the assessment was solely influenced by the data split, we only tested on the English (only English has sufficient gold data) and fine-tuned exclusively on the gold data, and (3) We tested all fine-tuned models in the first experiments on the long-text set and compositional set. We divided the compositional set into two subsets: substitution and extension, to assess the difficulty produced by these two operations.

For all experiments and models, uniform hyperparameters were employed, and the presented results are the average scores derived from three parallel experiments.[2]

4.3.1. Standard Test

Table 4 shows the results of the text-to-DRS parsing task. Across the four languages, both byT5 and DRS-MLM models stood out, with byT5 attaining 88.0 in German, slightly surpassing DRS-MLM's 87.1, and both models achieving the same F1 of 87.2 in Italian. However, in English and Dutch, DRS-MLM takes the lead with F1 91.5 and 85.5 respectively. mT5 and mBART closely follow, but their performance in Dutch is significantly weaker, possibly due to the limited Dutch data in their pre-training corpus.

Table 5 shows the results of DRS-to-text generation. ByT5 surpasses other models in all languages except for Dutch. Particularly in English, ByT5 achieves top scores with 71.9, 54.9, and 93.0 in three metrics, respectively. However, for the Dutch, DRS-MLM remains the superior model across these three metrics.

The standout performance of byT5 and DRS-MLM can be attributed to byte-level tokenization and specific pre-training, respectively. Unlike other tokenization methods, like Byte Pair Encoding (BPE, Sennrich et al., 2016), byT5's byte-level tokenization, which can be seen as character-level within our four target languages, results in a smaller dictionary and has the ability to handle unseen words. DRS-MLM employs several pre-training tasks on the PMB data, making the model better suited for the DRS data format. This advantage is most obvious when dealing with Dutch, which has the least training data among the four languages.

4.3.2. Systematic Split vs. Random Split

Figure 3 displays the distribution of word overlap rates between train and development/test sets under random and systematic split. The word overlap rate, defined in Equation 3, measures the word-level sentence similarity. According to the figure, the systematic word overlap distribution is further to the left than the random split, indicating that it has less overlap. And as outlined in Section 3, the systematic split does not simply reduce overlap by indiscriminately adding bias. It also guarantees that each set has a consistent length distribution, which can also be viewed as a semantic distribution to a certain extent. Therefore, in the case of PMB, a systematic split is a more effective method for dividing the dataset compared to the random split.

$$\text{overlap} = \frac{\text{sentence1} \cap \text{sentence2}}{\text{sentence1} \cup \text{sentence2}} \tag{3}$$

We further proved the advantage through experiments. The parsing and generation results under these two splits are shown in Table 6 and 7. The model's performance on the random split exceeds that on the systematic split for both tasks, suggesting the systematic approach presents more rigorous challenges.

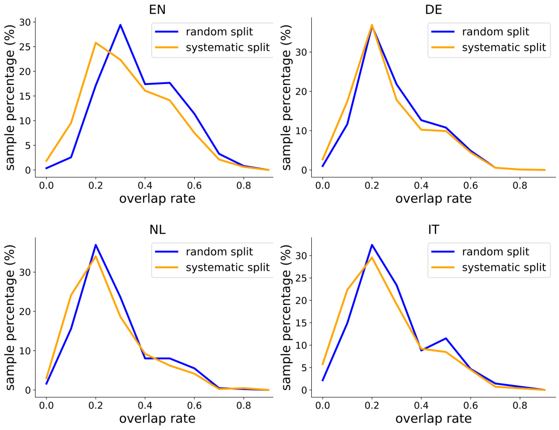


Figure 3: Distribution of word overlap rates between train and test sets in EN, DE, NL, IT. Lower overlap rates signify fewer words occurring in both train and test sets.

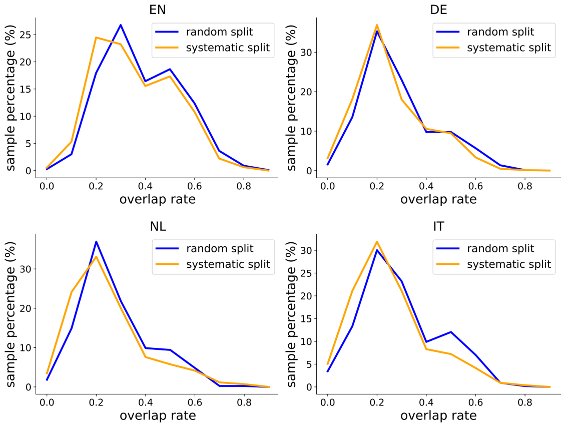


Figure 4: Distribution of word overlap rates between train and development sets in EN, DE, NL, IT.

4.3.3. Challenge Test Sets

The results of the models on the challenge test sets are shown in Tables 8 and 9. The performance on

[2] We provide the most recent experimental results for all test sets, available at https://pmb.let.rug.nl/models.php.

	English		German		Dutch		Italian	
Parser	F1	ERR	F1	ERR	F1	ERR	F1	ERR
LSTM	78.6	8.4	80.2	4.0	74.4	8.5	79.6	5.0
mT5	88.8	2.8	86.7	1.9	47.0	16.0	82.0	2.8
byT5	91.4	2.1	**88.0**	**0.7**	79.8	5.0	**87.2**	**0.7**
mBART	89.1	2.3	86.1	1.8	64.5	3.4	86.2	1.8
DRS-MLM	**91.5**	**1.5**	87.1	2.1	**85.5**	**2.0**	**87.2**	0.9

Table 4: Evaluation results for neural text-to-DRS parsing on the standard test sets of four languages. Note: ERR is the ill-formed rate (%) of generated DRSs that fail to transform into a graph structure.

	English			German			Dutch			Italian		
Generator	B	M	C	B	M	C	B	M	C	B	M	C
LSTM	33.8	32.4	72.5	24.9	25.4	67.1	19.0	21.6	63.2	28.2	24.7	72.2
mT5	69.9	53.4	92.8	47.8	37.5	84.8	11.3	15.2	63.6	48.8	36.3	86.0
byT5	**71.9**	**54.9**	**93.0**	**50.9**	**39.1**	**85.2**	41.8	34.2	82.1	**53.2**	**38.5**	**87.5**
mBART	51.8	43.5	88.1	40.8	33.4	79.9	38.1	32.0	80.6	45.8	34.5	84.7
DRS-MLM	67.5	52.4	92.2	47.6	36.6	84.4	**49.4**	**37.5**	**86.0**	46.3	34.2	86.3

Table 5: Evaluation results for neural DRS-to-text generation on the standard test sets of four languages. Note: B = BLEU; M = METEOR; C = COMET.

	Random split		Systematic split	
Parser	F1	ERR	F1	ERR
byT5	87.1	5.0	**83.5**	**6.0**
DRS-MLM	88.9	1.9	**87.3**	**4.1**

Table 6: Results of parsing under random and systematic split. Lower scores are marked.

	Random split			Systematic split		
Generator	B	M	C	B	M	C
byT5	66.1	52.2	91.7	**64.7**	**51.0**	**89.0**
DRS-MLM	65.8	51.4	91.7	**60.2**	**48.4**	**87.9**

Table 7: Results of generation under random and systematic split.

the long-text test set is significantly inferior, marked by a high incidence of ill-formed outputs[3]. The most pronounced drop is observed in ByT5, which shows a reduction of 86% compared to the standard test set. In the generation task, although truncation does not hugely impact on evaluation, the models still grapple with long sequences, reflecting decreases of at least 29.9, 11.9, and 16.2 across three metrics. Notably, neural models struggle with the long set, primarily because their tokenization significantly amplifies both input and output lengths. For example, while the average sentence lengths in the long set stand at 61 for text and 253 for DRS, these numebrs increase to 98 and 503 after BPE tokenization (mT5, mBART, and DRS-MLM) and even further to 410 and 1370 with character-level tokenization (ByT5). Obviously, these models can not handle such long sequences as effectively as the short sequences in the standard test.

For the compositional challenge set, it's crucial to note that all semantic components in the test sets were also in the training. Therefore, we expect near-perfect scores from the models. They perform well on the *compositional-substitution* set, showcasing their ability to learn and apply word meanings in known sentence structures. Among these models, byT5 performs the best with 93.1 F1 in parsing, while mT5 and DRS-MLM show similarly strong performance in generation. When testing on the *compositional-extension* set, the performance of the models dropped by around ten points in both tasks. Most parsing or generation errors were in the newly added parts in the texts, likely due to the introduction of more intricate sentence structures, especially compound predicate adjectives and attributive clauses, as shown in the examples in Table 3. The most frequent errors of the models are provided with examples in Appendix A.2.

5. Conclusion

Past performance of neural semantic parsers and meaning-to-text generators have been slightly in-

[3] SMATCH employs a hill-climbing technique to identify the optimal match, which may introduce inaccuracies when evaluating the output of the model for long texts (Opitz and Frank, 2022). In this case, the results for long texts should be considered as reference only.

	en-long		en-substitution		en-extension	
Parser	F1	ERR	F1	ERR	F1	ERR
LSTM	**43.7**	**19.2**	90.8	2.8	82.7	**3.5**
mT5	38.8	34.6	88.9	2.9	80.3	8.9
byT5	5.5	65.4	**93.1**	**0.5**	**84.8**	5.0
mBART	22.0	53.8	89.7	1.4	80.4	7.6
DRS-MLM	20.0	57.7	90.3	2.8	81.1	7.7

Table 8: Evaluation results for text-to-DRS parsing on the challenge test sets.

	en-long			en-substitution			en-extension		
Generator	B	M	C	B	M	C	B	M	C
LSTM	5.48	14.6	40.3	58.7	43.6	82.1	49.1	41.3	77.6
mT5	31.4	40.3	**76.6**	75.2	**55.6**	**92.7**	67.3	52.9	**90.0**
byT5	14.1	28.3	59.3	75.7	54.7	92.5	66.7	53.0	89.8
mBART	15.7	28.7	60.6	68.8	51.8	89.8	58.4	48.8	86.1
DRS-MLM	**32.6**	**40.5**	75.4	**76.0**	54.9	92.5	**69.4**	**53.2**	**90.0**

Table 9: Evaluation results for DRS-to-text generation on the challenge test sets.

flated (or at best, made the suggestion that these semantic computational tasks were close to being "solved") due to data leakage from training to test and non-representative test sets. At least, that is what our empirical study on the Parallel Meaning Bank showed. We created a more realistic assessment of performance by refining the data split and formulating challenge sets. A systematic split for the PMB yields a test set that is harder for semantic parsers and generators. The introduction of two further challenge sets, one with manually corrected longer documents and one with automatically derived compositional recombination using categorical grammar, are indeed way more challenging than the standard test set. Hence, semantic parsing and text-to-meaning generation can not be considered "solved" yet.

6. References

Mostafa Abdou, Artur Kulmizev, Vinit Ravishankar, Lasha Abzianidze, and Johan Bos. 2018. What can we learn from semantic tagging? In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4881–4889, Brussels, Belgium. Association for Computational Linguistics.

Lasha Abzianidze, Johannes Bjerva, Kilian Evang, Hessel Haagsma, Rik van Noord, Pierre Ludmann, Duc-Duy Nguyen, and Johan Bos. 2017. The Parallel Meaning Bank: Towards a multilingual corpus of translations annotated with compositional meaning representations. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 242–247, Valencia, Spain. Association for Computational Linguistics.

Lasha Abzianidze and Johan Bos. 2017. Towards universal semantic tagging. In *IWCS 2017 — 12th International Conference on Computational Semantics — Short papers*.

Lasha Abzianidze, Rik van Noord, Hessel Haagsma, and Johan Bos. 2019. The first shared task on discourse representation structure parsing. In *Proceedings of the IWCS Shared Task on Semantic Parsing*, Gothenburg, Sweden. Association for Computational Linguistics.

Lasha Abzianidze, Rik van Noord, Chunliu Wang, and Johan Bos. 2020. The parallel meaning bank: A framework for semantically annotating multiple languages. *Applied mathematics and informatics*, 25(2):45–60.

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 *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria. Association for Computational Linguistics.

Valerio Basile, Johan Bos, Kilian Evang, and Noortje Venhuizen. 2012. A platform for collaborative semantic annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 92–96, Avignon, France. Association for Computational Linguistics.

Johannes Bjerva, Barbara Plank, and Johan Bos. 2016. Semantic tagging with deep residual networks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3531–3541, Osaka, Japan. The COLING 2016 Organizing Committee.

C. Bonial, W. Corvey, M. Palmer, V.V. Petukhova, and H.C. Bunt. 2011. A hierarchical unification of lirics and verbnet semantic roles. In *Proceedings IEEE-ICSC 2011 Workshop on Semantic Annotation for Computational Linguistic Resources*, pages 1–7. Stanford University.

Johan Bos. 2008. Wide-coverage semantic analysis with Boxer. In *Semantics in Text Processing. STEP 2008 Conference Proceedings*, pages 277–286. College Publications.

Johan Bos. 2015. Open-domain semantic parsing with boxer. In *Nordic Conference of Computational Linguistics*.

Johan Bos. 2023. The sequence notation: Catching complex meanings in simple graphs. In *Proceedings of the 15th International Conference on Computational Semantics (IWCS 2023)*, pages 1–14, Nancy, France.

Shu Cai and Kevin Knight. 2013. 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, Sofia, Bulgaria. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. Cite arxiv:1810.04805Comment: 13 pages.

Yanai Elazar and Yoav Goldberg. 2018. Adversarial removal of demographic attributes from text data. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 11–21, Brussels, Belgium. Association for Computational Linguistics.

Federico Fancellu, Sorcha Gilroy, Adam Lopez, and Mirella Lapata. 2019. Semantic graph parsing with recurrent neural network DAG grammars. 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 2769–2778, Hong Kong, China. Association for Computational Linguistics.

Murhaf Fares, Stephan Oepen, Lilja Øvrelid, Jari Björne, and Richard Johansson. 2018. The 2018 shared task on extrinsic parser evaluation: On the downstream utility of English Universal Dependency parsers. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 22–33, Brussels, Belgium. Association for Computational Linguistics.

Qiankun Fu, Yue Zhang, Jiangming Liu, and Meishan Zhang. 2020. DRTS parsing with structure-aware encoding and decoding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6818–6828, Online. Association for Computational Linguistics.

Daniel Furrer, Marc van Zee, Nathan Scales, and Nathanael Schärli. 2020. Compositional generalization in semantic parsing: Pre-training vs. specialized architectures. *CoRR*, abs/2007.08970.

Kyle Gorman and Steven Bedrick. 2019. We need to talk about standard splits. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2786–2791, Florence, Italy. Association for Computational Linguistics.

Jonas Groschwitz, Shay Cohen, Lucia Donatelli, and Meaghan Fowlie. 2023. AMR parsing is far from solved: GrAPES, the granular AMR parsing evaluation suite. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10728–10752, Singapore. Association for Computational Linguistics.

Dag TT Haug, Jamie Y Findlay, and Ahmet Yıldırım. 2023. The long and the short of it: Drastic, a semantically annotated dataset containing sentences of more natural length. In *Proceedings of the 4th International Workshop on Designing Meaning Representations*, pages 89–98. Association for Computational Linguistics.

Da Huo and Gerard de Melo. 2020. Inducing universal semantic tag vectors. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3121–3127, Marseille, France. European Language Resources Association.

Mandar Juvekar, Gene Louis Kim, and Lenhart Schubert. 2023. Semantically informed data augmentation for unscoped episodic logical forms. In *15th International Conference on Computational Semantics*.

H. Kamp and U. Reyle. 1993. *From Discourse to Logic: Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Number pt. 2 in Developments in Cardiovascular Medicine. Kluwer Academic.

Robert T. Kasper. 1989. A flexible interface for linking applications to Penman's sentence generator. In *Speech and Natural Language: Proceedings of a Workshop Held at Philadelphia, Pennsylvania, February 21-23, 1989*.

Carina Kauf and Anna Ivanova. 2023. A better way to do masked language model scoring.

Najoung Kim and Tal Linzen. 2020. COGS: A compositional generalization challenge based on semantic interpretation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9087–9105, Online. Association for Computational Linguistics.

Brenden Lake and Marco Baroni. 2018. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *Proceedings of the 35th International*

Conference on Machine Learning, volume 80 of *Proceedings of Machine Learning Research*, pages 2873–2882. PMLR.

Brenden M. Lake and Marco Baroni. 2017. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International Conference on Machine Learning*.

Alon Lavie and Abhaya Agarwal. 2007. METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 228–231, Prague, Czech Republic. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Jiangming Liu, Shay B. Cohen, and Mirella Lapata. 2018. Discourse representation structure parsing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 429–439, Melbourne, Australia. Association for Computational Linguistics.

Jiangming Liu, Shay B. Cohen, and Mirella Lapata. 2019. Discourse representation parsing for sentences and documents. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6248–6262, Florence, Italy. Association for Computational Linguistics.

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.

Mitchell Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The penn treebank: Annotating predicate argument structure. In *Proceedings of the Workshop on Human Language Technology*, HLT '94, page 114–119, USA. Association for Computational Linguistics.

Tim O'Gorman, Michael Regan, Kira Griffitt, Ulf Hermjakob, Kevin Knight, and Martha Palmer. 2018. AMR beyond the sentence: the multi-sentence AMR corpus. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3693–3702, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Juri Opitz and Anette Frank. 2022. Better Smatch = better parser? AMR evaluation is not so simple anymore. In *Proceedings of the 3rd Workshop on Evaluation and Comparison of NLP Systems*, pages 32–43, Online. Association for Computational Linguistics.

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 on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Wessel Poelman, Rik van Noord, and Johan Bos. 2022. Transparent semantic parsing with Universal Dependencies using graph transformations. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4186–4192, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Nina Poerner, Hinrich Schütze, and Benjamin Roth. 2018. Evaluating neural network explanation methods using hybrid documents and morphosyntactic agreement. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 340–350, Melbourne, Australia. Association for Computational Linguistics.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The Penn Discourse TreeBank 2.0. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

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. *CoRR*, abs/1910.10683.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Julian Salazar, Davis Liang, Toan Q. Nguyen, and Katrin Kirchhoff. 2020. Masked language model scoring. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2699–2712, Online. 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, Berlin, Germany. Association for Computational Linguistics.

Anders Søgaard, Sebastian Ebert, Jasmijn Bastings, and Katja Filippova. 2021. We need to talk about random splits. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1823–1832, Online. Association for Computational Linguistics.

Mark Steedman. 1996. Surface structure and interpretation. In *Linguistic Inquiry*.

Rob van der Goot. 2021. We need to talk about train-dev-test splits. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4485–4494, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Rik van Noord. 2019. Neural boxer at the IWCS shared task on DRS parsing. In *Proceedings of the IWCS Shared Task on Semantic Parsing*, Gothenburg, Sweden. Association for Computational Linguistics.

Rik van Noord, Lasha Abzianidze, Antonio Toral, and Johan Bos. 2018. Exploring Neural Methods for Parsing Discourse Representation Structures. *Transactions of the Association for Computational Linguistics*, 6:619–633.

Rik van Noord, Antonio Toral, and Johan Bos. 2019. Linguistic information in neural semantic parsing with multiple encoders. In *Proceedings of the 13th International Conference on Computational Semantics - Short Papers*, pages 24–31, Gothenburg, Sweden. Association for Computational Linguistics.

Rik van Noord, Antonio Toral, and Johan Bos. 2020. Character-level representations improve DRS-based semantic parsing even in the age of BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4587–4603, Online. Association for Computational Linguistics.

Chunliu Wang, Huiyuan Lai, Malvina Nissim, and Johan Bos. 2023. Pre-trained language-meaning models for multilingual parsing and generation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5586–5600, Toronto, Canada. Association for Computational Linguistics.

Chunliu Wang, Rik van Noord, Arianna Bisazza, and Johan Bos. 2021a. Evaluating text generation from discourse representation structures. In *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)*, pages 73–83, Online. Association for Computational Linguistics.

Chunliu Wang, Rik van Noord, Arianna Bisazza, and Johan Bos. 2021b. Input representations for parsing discourse representation structures: Comparing English with Chinese. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 767–775, Online. Association for Computational Linguistics.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2022. ByT5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

A. Appendix

Appendix A.1 Pseudo-code for CCG recombination

Both substitution and extension operations begin with a standard pre-processing step: subtree set construction. This extracts all subtrees from the dataset's CCG derivation trees (For consistency, we treat leaves as subtrees with only the root). Substitution operation primarily involves randomly selecting subtrees, and then deleting and substituting them. The replacement subtree is chosen from the list in the first step. Extension operation involves forming child mappings and producing subtrees according to the mappings.

Algorithm 1 Extract Subtrees from CCG Trees

```
Variables:
$SubtreeList$ ← empty list
$AllCCGTrees$ ← CCG tree list

function ExtractSubs($node$, $currentPath$)
    if $node$ is null then return
    end if
    Add $node$ to $currentPath$
    if $node.left$ and $node.right$ are null then
        Add $currentPath$ to $SubtreeList$
    end if
    ExtractSubs($node.left$, $currentPath$)
    ExtractSubs($node.right$, $currentPath$)
end function

function SubtreesForTree($root$)
    ExtractSubs($root$, empty list)
return $SubtreeList$
end function

function SubtreesForTrees($AllCCGTrees$)
    for each $tree$ in $AllCCGTrees$ do
        SubtreesForTree($tree$)
    end for
return $SubtreeList$
end function
```

Algorithm 2 Substitution Operation

```
Variables:
$SubtreeList$ ← list of subtrees

function GetParent(tree, childNode)
    for each node $n$ in tree do
        if $n$.left = childNode or $n$.right = childNode then
            return $n$
        end if
    end for
    return null
end function

function DeleteAndAdd(tree, nodeToDelete)
    parent ← GetParent(tree, nodeToDelete)
    newSubTree ← randomly select from $SubtreeList$ with same root of nodeToDelete
    if parent.left = nodeToDelete then
        parent.left ← newSubTree
    else if parent.right = nodeToDelete then
        parent.right ← newSubTree
    end if
end function

function Substitute(tree)
    nodeToDelete ← randomly select a node from tree
    DeleteAndAdd(tree, nodeToDelete)
end function
```

Algorithm 3 Extension Operation

```
Variables:
$Subtrees$ ← list of subtrees
$ChildMap$ ← dictionary of children

function Traverse(node)
    if node is null then
        return
    end if
    if node.left then
        $ChildMap[(node, node.left)]$ ← node.right
    end if
    if node.right then
        $ChildMap[(node, node.right)]$ ← node.left
    end if
Traverse(node.left)
Traverse(node.right)
end function

function CreateSubtree(parent, left, right)
    parent.left = left
    parent.right = right
end function

function Extension(tree)
    $leaf$ ← RandomSelectLeaf(tree)
    if left then
        $newSubRoot$ ← CreateSubtree(leaf, leaf, $ChildMap[(leaf, leaf)]$)    ▷ To extend the node from right
    else
        $newSubRoot$ ← CreateSubtree(leaf, $ChildMap[(leaf, leaf)]$, leaf)    ▷ To extend the node from left
    end if
    choose the $newSubtree$ from $Subtrees$ according to $newSubRoot$
    replace $leaf$ with $newSubtree$
end function
```

Appendix A.2 Case Study

In this appendix, we present some wrong generations by byT5 model in the semantic parsing task. Additionally, the gold-standard text and DRS can also be seen as examples of the challenge sets.

Test set	Gold Text	Gold DRS	Generated
Standard	Mary called us.	female.n.02 Name "Mary" call.v.03 Agent -1 Time +1 Co-Agent +2 time.n.08 TPR now person.n.01 Sub speaker	female.n.02 Name "Mary" call.v.03 Agent -1 Time +1 Theme +2 time.n.08 TPR now person.n.01 Sub speaker
Long Text	Recent studies show that children who do not get enough sleep tend to have some emotional problems as well as weight gain later in life. As VOA's Melinda Smith reports, the research seems to blame the parents.	recent.a.02 AttributeOf +1 study.n.01 show.v.02 Proposition >1 Experiencer -1 Time +1 time.n.08 EQU now CONTINUATION <0 child.n.01 tend.v.01 Agent -1 Time +1 Topic +2 time.n.08 EQU now have.v.01 Pivot -3 Theme +3 Theme +7 emotional.a.03 AttributeOf +1 problem.n.01 entity.n.01 Sub -1 Sub +2 weight.n.01 gain.n.01 Theme -1 later.r.01 EQU -6 life.n.01 NEGATION <1 time.n.08 EQU now get.v.01 Pivot -12 Time -1 Theme +2 enough.a.01 AttributeOf +1 sleep.n.01 CONTINUATION <3 agency.n.01 Name "VOA" female.n.02 Name "Melinda Smith" PartOf -1 report.v.01 Agent -1 Time +1 time.n.08 EQU now CONTINUATION <1 research.n.01 seem.v.01 Experiencer -1 Time +1 Stimulus +2 time.n.08 EQU now blame.v.01 Agent -3 Theme +1 person.n.01 Role +1 parent.n.01	recent.a.01 AttributeOf +1 study.n.04 show.v.04 Proposition >1 Experiencer -1 Time +1 time.n.08 EQU now CONTINUATION <0 child.n.01 NEGATION <1 time.n.08 EQU now get.v.01 Pivot -2 Time -1 Theme +2 enough.a.01 AttributeOf +1 sleep.n.01 tend.v.01 Agent -4 T
Substitution	Hungarian prisoners broke out of jail.	country.n.02 Name "Hungary" person.n.01 Location -1 Role +1 prisoner.n.01 break_out.v.03 Theme -2 Time +1 Source +2 time.n.08 TPR now jail.n.01	country.n.02 Name "Hungary" person.n.01 Source -1 Role +1 prisoner.n.01 break_out.v.01 Source -2 Time +1 Theme +2 time.n.08 TPR now jail.n.01
Extension	Mr. Smith who worked on that project asked Jane to marry him.	mr.n.01 male.n.02 Name "Smith" Title -1 work.v.01 Agent -1 Time +1 Theme +2 time.n.08 TPR now project.n.01 ask.v.02 Agent -4 Time +1 Recipient +2 Topic +3 time.n.08 TPR now female.n.02 Name "Jane" marry.v.01 Agent -1 Co-Agent +1 male.n.02 ANA -8	mr.n.01 male.n.02 Name "Smith" Title -1 work.v.02 Agent -1 Time +1 Theme +2 time.n.08 TPR now project.n.01 ask.v.02 Agent -4 Time +1 Patient +2 Result +3 time.n.08 TPR now female.n.02 Name "Jane" marry.v.01 Agent -1 Co-Agent +1 male.n.02 ANA -5

Table 10: Four examples in different test sets.

Author Index