NEJLT

Northern European Journal of Language Technology

www.nejlt.org

Volume 8, December 2022
ISSN 2000-1553

NEJLT Editorial Team 2022

Foreword to NEJLT Volume 8, 2022

Leon Derczynski, ITU Copenhagen, Denmark; ld@itu.dk

Abstract An introduction to the Northern European Journal of Language Technology in 2022

1 Introduction

This introduces the second of two volumes of the Northern European Journal of Language Technology, or NEJLT, under a revised remit. We have expanded the journal to carry excellent peer-reviewed works on natural language processing and computation linguistics from across the world. This has been a success and it is a delight to present these volumes 7 and 8.

Of note, in addition to what we have come to expect of NLP venues, the also offers:

- An editorial board of leading action editors from institutes across the world;
- A focus on global languages;
- The adoption of specific paper types, each with their own review form, to increase the chances that authors get a review that suits them, and that review does not become overfit to the more common types of submitted papers (Derczynski and Bender, 2021);
- Moves to include NEJLT in the ACL Anthology;
- Welcoming of reviews from previous venues to be submitted with papers, to reduce the number of rounds needed for decisions;
- The addition of a "letter" format submission, for short contributions with rapid review.

This has taken a significant of work, especially from action editors and reviewers, and much trust, especially on behalf of authors submitting their hard work to a newer venue. I am extremely grateful to those who have placed their energy and willpower into pushing NEJLT to where it is now.

In 2022, NEJLT received seventeen submissions and published eight manuscripts. Of these eight, the languages covered by their research included Abui, Algerian Judeo-Arabic, Bardi, Basque, Chintang, Dutch, Finnish, German, Greek, Haiki, Hausa, Hungarian, Ik, Indonesian, Italian, Japanese, Korean, Latin, Lezgi, Mandarin, Matsigenka, Meithei, Nuuchahnulth, Old Javanese, Polish, Russian, Spanish, Swedish, Tsova-Tush, Turkish, Wambaya, Yup'ik, and English. Publication authors held affiliations in Germany, Israel, Norway, Sweden, the UK, and the USA. So if one is in any doubt as to NEJLT's natural as not a Northern European journal but a global one, please look no further.

The journal continues to offer fast review, free submission, publication, and reading, and a top-quality board. We look forward to fair and fast review of even more exciting papers in 2023. To learn more, please visit www.nejlt.org.

References

Derczynski, Leon and Emily M Bender. 2021. Towards better interdisciplinary science: Learnings from COLING 2018. Technical report, IT University of Copenhagen.

Task-dependent Optimal Weight Combinations for Static Embeddings

Nathaniel R. Robinson, Carnegie Mellon University, USA nrrobins@cs.cmu.edu
Nathaniel Carlson, Brigham Young University, USA natec18@byu.edu
David R. Mortensen, Carnegie Mellon University, USA dmortens@cs.cmu.edu
Elizabeth Ann Vargas, Brigham Young University, USA elizag17@byu.edu
Thomas Fackrell, Brigham Young University, USA tfac1997@byu.edu
Nancy Fulda, Brigham Young University, USA nfulda@cs.byu.edu

Abstract A variety of NLP applications use word2vec skip-gram, GloVe, and fastText word embeddings. These models learn two sets of embedding vectors, but most practitioners use only one of them, or alternately an unweighted sum of both. This is the first study to systematically explore a range of linear combinations between the first and second embedding sets. We evaluate these combinations on a set of six NLP benchmarks including IR, POS-tagging, and sentence similarity. We show that the default embedding combinations are often suboptimal and demonstrate up to 12.5% improvements. Notably, GloVe's default unweighted sum is its least effective combination across tasks. We provide a theoretical basis for weighting one set of embeddings more than the other according to the algorithm and task. We apply our findings to improve accuracy in applications of cross-lingual alignment and navigational knowledge by up to 15.2%.

1 Introduction

Static word embeddings are used in a broad range of NLP applications, including conversational gameplay (Andrus and Fulda, 2020), text categorization (Minaee et al., 2021; Mitra et al., 2016), translation (Sabet et al., 2020; Jansen, 2017; Pourdamghani et al., 2018), affordance detection (Fulda et al., 2017a), and semantic analysis (Hamilton et al., 2016). In addition to using static embeddings directly, researchers often combine them with contextualized models or use them for embedding initialization of downstream tasks (Kocmi and Bojar, 2017) such as summarization (Lin et al., 2021) and neural machine translation (Qi et al., 2018). The persistence of static embeddings is due in part to their ease of use and low computational requirements. Rather than needing a forward pass through a neural network to embed each word, pre-trained embeddings can be stored in memory and retrieved with complexity $O(1)$.[1] §4.2 outlines more advantages of static embeddings.

[1]This is assuming a default Python or Java hash map. The worst case would be $O(n)$ with vocabulary size n, in the case of a trivially slow hash map, which is still well under transformer-based embedding retrieval complexity.

We propose an augmentation of three popular embedding methods (word2vec skip-gram, GloVe, and fastText). Word2vec skip-gram (Mikolov et al., 2013a) is a neural word context predictor, GloVe (Pennington et al., 2014) is a log-bilinear model that includes global context information with a co-occurrence matrix, and fastText (Bojanowski et al., 2017) incorporates sub-word information via character n-grams with a skip-gram objective to expedite training and handle unseen words. More details about these algorithms are in §2. Each of them produces two separate embedding sets ("target" and "context", see Figure 1) that we combine in previously unexplored ways. We show that these typically unexplored target and context combinations reveal much about embedding effectiveness. Our key contributions are as follows:

1. We provide a theoretical and empirical analysis of static embedding performance across weighted linear combinations of embedding sets ("target" and "context").
2. We generate 126 embedding sets from 6 corpora and show that the default target/context combination for each embedding algorithm is often sub-

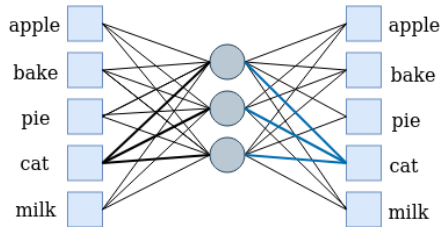


Figure 1: Illustration of "target" vectors (black lines, left) and "context" vectors (blue lines, right) produced by the word2vec skip-gram algorithm. Similar embedding pairs exist for fastText and GloVe. These are sometimes called "in" and "out" weights, respectively.

optimal.

3. We demonstrate improvements on analogies, textual similarity, IR, POS-tagging, cross-lingual alignment, and robotics navigation via embedding combinations and provide best practice recommendations.

We demonstrate up to 12.5% improvements over baseline performance on a diverse set of NLP benchmarks by combining target and context vectors. (See §5.) We analyze embeddings statistically and show that (1) word2vec target vectors encode better word-to-word relationships while context vectors are better suited for bag-of-words representations, (2) GloVe default vectors perform well on tasks for which they were tuned but under-perform generally, and (3) fastText target vectors' sub-word encodings are useful in many tasks but counterproductive for bag-of-words representations. (See §6.) Finally, in §7 we employ our methods practically to improve performance on MUSE (Lample et al., 2018) cross-lingual alignment by 0.69-1.56% and navigational robotics benchmarks by up to 15.2%

2 Background

We overview three common static embedding algorithms: word2vec skip-gram with negative sampling, fastText, and GloVe. Each of these algorithms has the same output: a set of embedding vectors, each corresponding to a word in a vocabulary. We outline applications that employ these different algorithms, but note that any task that uses vectors from one of these algorithms could just as feasibly use the vectors from another of them.

Mikolov et al.'s (2013a) **word2vec skip-gram** model learns embeddings as a neural regression problem: predicting each word's context. Each word in question is called the *target*, and its neighbors are called *context*. The model learns two sets of embeddings, corresponding to target and context words, though only the first is typically used. Word2vec is employed in many tasks, including measurement of MWE candidates (Pickard, 2020) and epidemic-related twitter stream classifications (Khatua et al., 2019). In our experiments we used skip-gram with negative sampling (rather than Mikolov et al.'s CBOW model) because of its comparability to GloVe and fastText. (See §3.) Throughout the text we refer to this algorithm simply as "word2vec."

The **fastText** algorithm (Bojanowski et al., 2017) integrates sub-word information into the skip-gram framework. It embeds character n-grams, and a word's embedding is the sum of its sub-word vectors. The context vectors are not composed of sub-words. Many applications use fastText, including hyperbolic word representations (Zhu et al., 2020) and low-resource sentence similarity (Khalid et al., 2021; Akhtar et al., 2017).

Popular **GloVe** embeddings (Pennington et al., 2014) are used for sarcasm detection (Khatri and P, 2020), emotion detection (Gupta et al., 2021), and lexical semantic analysis (Jain, 2020). GloVe's log-bilinear model learns two embeddings from word co-occurrence. By default, most public GloVe implementations sum the embeddings evenly. However, our study shows that this unweighted sum often does not maximize performance. (See §5.)

Our work challenges the assumption that the default combinations or selections of target and context vectors (*target only* for word2vec and fastText and *an unweighted sum* for GloVe) are optimal for any task, or even across tasks in general. We methodically explore a spectrum of target/context combinations for each algorithm and show that the default embedding selection is often not the best.

Although not directly studied here, other static embedding algorithms such as ConceptNet Numberbatch (Speer et al., 2017), hyperbolic word embeddings (Zhu et al., 2020), and word2vec-CBOW (Mikolov et al., 2013a) exist and merit study.

2.1 Contextual Embedding

The uses of static embeddings overlap with contextualized embedding models such as BERT (Devlin et al., 2019), ELMo (Peters et al., 2018), BART (Lewis et al., 2020), and others (Liu et al., 2019b; Robinson et al., 2021). These networks are adaptable to a variety of NLP tasks, from translation evaluation (Yuan et al., 2021; Zhang et al., 2019) to semantic tagging (Liu et al., 2019a). Some researchers have scrutinized them for consuming too many resources and lacking interpretability (Bender et al., 2021; Brown et al., 2020; Strubell et al., 2019). Static embeddings are employed instead in many practical NLP tasks because they are fast, computationally inexpensive, and intuitive.

Static embeddings are particularly suited to tasks that restrict predictions to a candidate set, such as word

analogies, since embeddings from these smaller models have a defined vocabulary that can be queried in nearest-neighbor search. Dufter et al. (2021) verified this trend for question answering and advocated for use of static embeddings because of their low computational cost: "'green' baselines are often ignored, but should be considered when evaluating resource-hungry deep learning models."

2.2 Prior Investigations of Context and Target Combinations

We are not the first to combine target and context embeddings. As mentioned in §2, GloVe implementations (Pennington et al., 2014) sum them by default. Nalisnick et al. (2016) used target vectors for queries and context vectors for documents in information retrieval (IR). They did not explore summing the two vector sets however.

Fulda and Robinson (2021) explored concatenating and summing word2vec skip-gram target and context vectors for analogies and sentence similarity. They found that with sufficiently large training corpora, target-context sums can outperform target embeddings (the default) and target-context concatenations. Their analysis reveals a theoretical advantage for summed embeddings in analogy tasks with dot-product-based similarity metrics: with target-context sums, the dot product between vectors for words a and b is $(t_a + c_a)^T(t_b + c_b) = t_a^T t_b + t_a^T c_b + c_a^T t_b + c_a^T c_b$, where t and c are target and context vectors. This is significant because, as Nalisnick et al. (2016) point out, the outer terms of this expression encode paradigmatic relations (e.g. *mason* and *carpenter*), while the inner terms encode syntagmatic relations (e.g. *mason* and *stone*). These different relations are relevant in analogies like *mason : stone :: carpenter : wood*.

Our work expands on and differs from these examples in four main ways: (1) we systematically explore *multiple* combinations of target and context vectors (e.g. 80% target added to 20% context, which we show in §6.1 may be conceptually closer to a true sum), (2) we apply these changes uniformly across word2vec, GloVe, and fastText on multiple training corpora and a variety of NLP tasks, (3) we provide a theoretical analysis of the properties of target and context vectors by task type, and (4) we identify a set of best practices and recommended embedding combinations for practitioners applying these algorithms in the wild.

3 Theoretical Motivation

This section shows that while the target and context vectors produced by the GloVe algorithm are fundamentally equivalent to each other, the same does not hold true in the cases of word2vec and fastText. This insight informs and motivates our analysis of various context/target embeddings in §6.

Word2vec Skip-gram Training The skip-gram model processes a corpus, considering each word as a target and its surrounding words within a window size w as context. For each step, the weights are updated for two objectives:

- **Objective 1**: Maximize the dot product between the target embedding for the target word and the context embeddings for its neighbors
- **Objective 2**: *Negative sampling*, or minimize the dot product between the target embedding for the target word and the context embeddings of random words

Theorem 1 shows that **objective 1** cannot account for any difference between target and context space. We outline terminology for the theorem below.

Terminology for Theorem 1: Let the operation $close(v_1, v_2)$ indicate an action: that of model weight updates to increase the dot product between vectors v_1 and v_2. All of the model's actions for **objective 1** as it processes the corpus once are

$$\{\{p(|j-i|)close(t_i, c_j)\}_{j=i-w, j\neq i}^{i+w}\}_{i=0}^{M-1} \quad (1)$$

where t_i and c_i are the target and context embedding for the word appearing in the ith position of the corpus, and M is the corpus length. The function $p(\ell)$ is a Bernoulli c.d.f. that returns 0 with probability $\frac{(\ell-1)}{w}$. This expresses how the skip-gram model probabilistically drops **objective 1** actions for context words far from the target. We require that $close(v_i, v_j)$ represent no action if either i or j is below zero or greater than $M-1$. Note that the arguments of $close$ may be commuted.

Given Theorem 1, all of the differences between target and context structure are due to **objective 2**. We show how common word context vectors behave like magnets for target vectors to cluster around, because of **objective 2** action distribution.

Let $far(v_1, v_2)$ indicate the action of weight updates to decrease the dot-product between v_1 and v_2 (the opposite of $close$). All **objective 2** actions from one reading of the corpus are

$$\{\{far(t_i, c_{x_j})\}_{j=0}^{5r-1}\}_{i=0}^{M-1} \quad (2)$$

where 5 is the set number of negative samples per word, r is a random integer $1 \leq r \leq w$ that depends on the output of p in Equation 1, and each c_{x_j} denotes the context vector for a word at index x_j drawn randomly from a distribution X of square-rooted word frequencies. Let X_{True} be the distribution of actual word frequencies from the corpus. When sampling from square-rooted word frequencies X, the most common words in

Theorem 1. ***Objective 1*** *does not cause any differences between target and context vector construction.*

Proof. Define $f(v_i, v_j) := p(|j-i|)close(v_i, v_j)$ for convenience, and note $f(v_i, v_j) = f(v_j, v_i)$. The set of total **objective 1** actions performed by the model as it reads the corpus once is then

$$\begin{aligned}
&\{\{f(t_i, c_j)\}_{j=i-w, j\neq i}^{i+w}\}_{i=0}^{M-1} \\
&= \{\{f(t_i, c_j)\}_{i-w\leq j\leq i-1}, \{f(t_i, c_j)\}_{i+1\leq j\leq i+w}\}_{i=0}^{M-1} \\
&= \{\{f(c_i, t_j)\}_{j-w\leq i\leq j-1}, \{f(c_i, t_j)\}_{j+1\leq i\leq j+w}\}_{j=0}^{M-1} \\
&= \{\{f(c_i, t_j)\}_{j-w\leq i\leq j-1}\}_{j=0}^{M-1} \cup \{\{f(c_i, t_j)\}_{j+1\leq i\leq j+w}\}_{j=0}^{M-1} \\
&= \{\{f(c_i, t_j)\}_{i+1\leq j\leq i+w}\}_{i=-w}^{M-2} \cup \{\{f(c_i, t_j)\}_{i-w\leq j\leq i-1}\}_{i=1}^{M-1+w} \\
&= \{\{f(c_i, t_j)\}_{i+1\leq j\leq i+w}\}_{i=0}^{M-1} \cup \{\{f(c_i, t_j)\}_{i-w\leq j\leq i-1}\}_{i=0}^{M-1} \\
&= \{\{f(c_i, t_j)\}_{i-w\leq j\leq i-1}\}_{i=0}^{M-1} \cup \{\{f(c_i, t_j)\}_{i+1\leq j\leq i+w}\}_{i=0}^{M-1} \\
&= \{\{f(c_i, t_j)\}_{j=i-w, j\neq i}^{i+w}\}_{i=0}^{M-1},
\end{aligned}$$

which is the original expression with the roles of t and c vectors reversed. Therefore c and t are reversible without changing the total **objective 1** actions performed by the model. I.e. their roles are identical in this process. □

the corpus will be less frequent, and the least common words will be more frequent, than when sampling from X_{True}. In Equation 1, we have $t_i, c_j \sim X_{\text{True}}$. However, in Equation 2, we have $t_i \sim X_{\text{True}}$ but $c_{x_j} \sim X$. This means that the context vectors c corresponding to frequent words will appear more often in the *close* function and less often in far (and context vectors for *infrequent* words will appear more often in far than in *close*). Therefore the target vectors will be biased to be more similar to the context vectors of high frequency words.

A brief analysis shows this. Let S_{100} be the 100 most common words in a corpus from the vocabulary V. For word2vec embeddings trained on three corpora, (Web Scraped, WikiReddit, and Wikipedia, described in §4.1) we calculated the cosine scores between all target vectors and the 100 context vectors for the words in S_{100}, $\text{scores}_1 = \{\cos(t_v, c_s) : v \in V, s \in S_{100}, v \neq s\}$ and the corresponding scores where target vectors came from S_{100}, $\text{scores}_2 = \{\cos(c_v, t_s) : v \in V, s \in S_{100}, v \neq s\}$. For each corpus, over 90% of the 100 highest scores in $\text{scores}_1 \cup \text{scores}_2$ were from scores_1, indicating that target vectors are clustered around common-word context vectors (more so than context vectors are clustered around common-word target vectors).

fastText vectors are trained using this same paradigm but with an additional difference between the embedding sets that is likely more important in applications: target vectors are composed by summing subword embeddings, while context vectors are not.

GloVe vectors are constructed differently than word2vec or fastText. Their training objective $(\sum_{i,j}^{V} f(X_{i,j})(t_i^T c_j + b_i + \tilde{b}_j - log(X_{i,j}))^2)$ is log-bilinear for target and context vectors. Thus there is no difference between target and context vector construction, short of random initialization. Analyses of vector space clustering and magnitude for GloVe, including the analysis described in the previous paragraph and statistics outlined in Table 4, reveal no notable differences between GloVe target and context distributions.

4 Methodology

We conduct experiments to answer three core questions: (1) How does task performance vary across linear combinations of target/context vectors, and do the default settings work generally well? (2) Is the pattern of performance (as a function of target/context weightings) similar in all three embedding algorithms? If not, what are the differences? (3) Are optimal weighting schemes data- and task-specific? If so, to what extent?

To answer these, we trained target and context embeddings for each of three algorithms (word2vec skip-gram, GloVe, fastText) on six corpora. We then produced more embeddings by combining each pair of target t and context c as follows:[2] $.8t + .2c$, $.6t + .4c$, $t + c$, $.4t + .6c$, and $.2t + .8c$. We refer to these combinations respectively as 80:20-sum, 60:40-sum, true sum, 40:60-sum, and 20:80-sum, in addition to target and context vectors alone. We generated 126 embedding sets total (six training corpora × three embedding algorithms × seven weighted sums). This spread of linear combinations has not been studied previously.

We insist that the target and context weights sum to 1 for the sake of uniformity and clarity in drawing

[2] $t + c$ is a stand-in for $.5t + .5c$, since our NLP tasks rely on magnitude-agnostic cosine similarity. We use $t + c$ to compare directly with existing methods.

conclusions from our experiments. Since our NLP applications employ magnitude-agnostic cosine distance as a similarity metric, allowing the weights to range from 0 to 1 is largely equivalent to letting them range from 0 to N for any positive real number N. We set $N = 1$ across all experiments to allow fair comparisons of weight values. It is worth noting that weights could also be negative, a possibility that is beyond the scope of our current study but that could be explored in future work.

Linear combinations are a form of summation. Summation is an established method for combining vector information, as discussed in §2.2. The fastText algorithm relies on sums of subword target vectors (Bojanowski et al., 2017). GloVe vectors are often target-context sums by default (Pennington et al., 2014). Target and context vectors ought to be reasonably well aligned for summation, since the objective of a static embedding model is to increase the dot products between target and context vectors, and the dot product is based on component-wise multiplication. This provides a theoretical motivation for our focus on target-and-context vector summation. Note, however, that summation is not the only possible way to combine target and context embeddings. As discussed in §2.2, Fulda and Robinson (2021) explored concatenating the embeddings instead. However, they found that summation yielded better results, a conclusion that they verified both theoretically and experimentally.

Despite the theoretical basis for their alignment, target and context vectors may not be perfectly aligned for summation in practice. One could employ a more intricate approach to align them, such as a linear mapping or an encoder/decoder reconstruction method such as MUSE (Lample et al., 2018). This is a potentially promising area of future research.

Note that our experimentation over a range of seven different target/context weighting values is comparable to performing a course grid search for this parameter in each task. Another more computationally expensive and task-specific method to tune this parameter is meta-learning. Our objective in our main experiments is to learn generalizable principles across a variety of tasks, which may be of value to other researchers, for which purpose we determined that this grid search approach would suffice. However, we do explore an application of optimizing the weighting via differentiation. See §7.2 for analysis.

Training hyperparameters We used embedding dimension 300. For **GloVe**: we used window size 10, minimum word count 5, and 25 training iterations. These have been shown to achieve optimal results (Pennington et al., 2014). The "minimum word count" mentioned is minimum frequency for a word in the corpus to include it in the model's vocabulary. For **fastText**: we used window size 10, minimum word count 5, 5 training epochs, and 3-6 character n-grams, as standard (Bojanowski et al., 2017). For **word2vec**: we used window size[3] 5 and 3 epochs, as recommended by Fulda and Robinson (2021). Due to the Web Scraped corpus' size and computing restraints, we opted for minimum word count 100.

4.1 Training Corpora

Our six training corpora are in Table 1. The Web Scraped corpus was generated to imitate the unreleased WebText data used to train OpenAI's GPT-2 (Peterson, 2019; Radford et al., 2019). The Wikipedia corpus is a collection of all text on Wikipedia from 2004. The WikiReddit dataset is the concatenation of the Wikipedia corpus with text from Reddit. The Toronto Books Corpus contains 11,038 books collected by the University of Toronto (Zhu et al., 2015). The smallest corpus consists of classic books from Project Gutenberg (Lahiri, 2014). Spanish Wikipedia was the dump from October 20, 2021 collected using the WikiExtractor tool (Attardi, 2015).

We had the majority of these corpora on hand and used them to reduce computational expense. Though the NER task could potentially have benefited from using newer corpora, such as a more recent download of English Wikipedia, we did not see a clear theoretical impact from using newer corpora on the other evaluation tasks. We provide a brief analysis of the interaction between embedding performance and corpus choice in 6.4. An in-depth analysis of this interaction is outside the scope of this paper, but we recommend it as an area of further study.

In Table 2 we show the vocabulary sizes for the embedding sets trained using the three embedding algorithms and the six training corpora in our experiment set.

Corpus	Size	Tokens
Web Scraped	59.0 GB	9.6B
WikiReddit text	21.0 GB	4.1B
Wikipedia text	16.7 GB	2.8B
Toronto Books	4.6 GB	984M
Classic Books	20.3 MB	<1M
Spanish Wikipedia	4.5 GB	667M

Table 1: Training corpora (five English, one Spanish). Novel corpora will be released upon acceptance.

[3]Our word2vec implementation denotes unidirectional window size. This value is equivalent to the bi-directional window size 10 for fastText and GloVe.

	word2vec	fastText	GloVe
Web Scraped	408K	3.36M	3.36M
WikiReddit text	530K	2.51M	2.51M
Wikipedia text	432K	1.95M	1.95M
Toronto Books	92.3K	315K	315K
Classic Books	5.69K	36K	36K
Spanish Wikipedia	347K	2.24M	2.24M

Table 2: Vocabulary sizes for each embedding set trained on each of the corpora, using the hyperparameters delineated in §4 (using 3 significant figures)

4.2 Evaluation Tasks

We evaluated each embedding set on six NLP tasks chosen to represent a broad sampling of static embedding uses, conducting over 650 evaluations. We describe task details below for reproduction.

We employed two analogy question evaluations. **The Google Analogy Test Set** (Mikolov et al., 2013c,b) is a common embedding evaluation benchmark. It contains 19,544 analogy questions in 14 categories: six semantic (e.g. family relationships, countries and capitals) and eight grammatical (e.g. adjectives with superlatives). GloVe vectors were tuned for this benchmark. **Turney's (2006) set of SAT questions** was used by Fulda and Robinson (2021) to evaluate static embeddings. It contains 374 analogy questions with semantic relationships like the mason/carpenter example in §2.2.

Selection of analogy candidates: Given analogy A : B :: C : D, in the Google test, embeddings predict D given A, B, and C from $\hat{d} = \arg\max_{d' \in S} \cos(b - a + c, d')$, where a, b, and c are vectors corresponding to their respective words, and S is the set of all vectors in the embedding. *cos* denotes cosine similarity. In the SAT test, embeddings predict C and D given A and B from $\hat{c}, \hat{d} = \arg\max_{(c',d') \in S} \cos(b - a, d' - c')$, where S is a set of four multiple-choice candidate pairs (c_i, d_i). To illustrate more intuitively, say we have the example analogy *mason : stone :: carpenter : wood*. In the Google test paradigm, our task is to predict "wood" given "mason," "stone," and "carpenter." And we do this by finding the word in our vocabulary whose vector is closet to $v_{stone} - v_{mason} + v_{carpenter}$. In the SAT paradigm, we are given the pair "mason" and "stone," along with a list of multiple-choice options, each containing a pair of words and one of which contains the pair "carpenter" and "wood." We select the option whose pair-wise difference vector is closest to the difference vector for the given pair. Because of the high propensity of esoteric words in SAT analogies, we skip SAT questions with out-of-vocabulary words.

SemEval 2013 (Wilson et al., 2013) is a sentence textual similarity (STS) set of 1,379 sentence pairs with human-given similarity scores. We sum word vectors to obtain sentence vectors, then measure how well pairwise cosine similarity correlates with gold similarity using Spearman's rho.

We adapt Nalisnick et al.'s (2016) **IR method**, the Dual Embedding Space Model (DESM). We collected 36,701 queries and 3.2 million documents from Ni (2015). Each query is mapped to a list of 100 relevant documents with relevance scores. For query-document similarity we calculate a modified DESM score $\text{DESM}(Q, D) = \frac{1}{|Q|} \sum_{q \in Q} \frac{q^T \overline{D}}{||q|| ||\overline{D}||}$, where Q and D are matrices of vectors for the query and document words,[4] respectively, and $\overline{D} = \frac{1}{|D|} \sum_{d \in D} d$. We rate embeddings by average Spearman's Rho correlation between DESM scores and ground truth document relevance scores across queries.

Our method for **POS tagging** is adopted from Premjith (2019). We predict eight POS categories (noun, adjective, adverb, adposition, determiner, pronoun), using the highest-performing of five classifiers: K-neighbors, decision tree, random forest, multi-layer perceptron, and Gaussian naive Bayes. These are the models and parameters used by Premjith (2019). In recent years, other models such as LSTM and encoder/decoder models have been commonly used for POS-tagging. We opted to keep the NLP applications we gathered as close to their original form as possible, for the sake of uniformity, and as such we do not augment the model list with these additional architectures. We acknowledge that a POS-tagging embedding evaluation employing LSTM and seq2seq models could be worthwhile in future studies. In all cases, embedding vectors were used as input to the tagging models. We evaluate both classifiers and embeddings with POS prediction weighted-averaged F1 score.

Our **cross-lingual alignment task** is adapted from Jansen (2017). We train a transition matrix between English and Spanish Wikipedia embeddings on a 2894-word English-Spanish dictionary with a 64-16-20 train-dev-eval split. We train for 10 epochs with learning rate .001 and the Adam optimizer (Kingma and Ba, 2015). We rate embeddings by validation accuracy. In each test, the settings of Spanish and English embedding training are identical (same embedding algorithm and target/context combination).

Tasks summary: This set of evaluations contains both common embedding benchmarks and practically relevant tasks. Static embeddings are particularly suited to multiple of these tasks because they involve context-less word relationships and proximity searches across stored embedding sets.

[4] Excluding stop words such as "and," "that," or "may", as defined by SpaCy model `en_core_web_sm`. See https://spacy.io/models/en.

5 Results

Our results show that the default settings of each algorithm for combining target and context vectors do not always perform best, and often perform worst, on NLP tasks. Figure 2 shows the performance of 7 target/context combinations across the 5 English corpora and 3 embedding algorithms.

In summary, for **word2vec**, context vectors perform best on IR, but target vectors are best on SAT and Google analogy tasks and cross-lingual alignment, and summed vectors excel in STS. **GloVe** vectors exhibit two major trends. Summed vectors perform best in STS but worst in other tasks: SAT analogies, POS tagging, and IR. **fastText** target vectors perform best on Google analogies and POS tagging, but in IR and cross-lingual alignment, summed vectors excel.

Table 3 shows the percentage improvement from tuning the target/context weighting over default weighting for each embedding algorithm and evaluation task. These values represent the improvement from the highest performance using the default target/context setting to the highest performance after our search over linear combinations. We see the largest performance increase, 12.5%, in the case of word2vec on the STS task.

Patterns across target and context combinations are dependent on the NLP task and algorithm. This suggests that tuning the target/context summation weight (rather than using defaults) can improve performance markedly. For example, GloVe's default true sum does well on the STS task but under-performs on SAT analogy, POS, and IR. Word2vec and fastText's default target embeddings perform well on the Google Analogy and POS tasks but under-perform on the IR task.

Performance trends are generally consistent across our five training corpora. There are some notable exceptions to this generality, which we discuss in §6.4.

Average Effect of Target/Context Weighting Across Corpora and Tasks Figure 3 shows the average performance of each algorithm across all corpora and tasks. Results suggest fastText performs optimally with a 20:80 target/context combination rather than the default setting of 100% target. GloVe performance is highest at 80:20 and 20:80. These results suggest that a 20:80 or 80:20 combination of target and context may be an advantageous default for future embedding sets, especially in settings where hyperparameter tuning is not possible. (E.g. because embeddings are pre-trained or due to computational constraints.) In §6 we analyze results, and in §7 we present practical applications for these observations.

6 Analysis

This section provides theoretical backing for the observed performance of target vs. context vectors on specific types of tasks. We analyze the advantages of using weighted target and context combinations for specific use cases and offer recommendations for best practices in static embedding research.

6.1 Word2vec Analysis

Word2vec target vectors outperform their context counterparts in analogy tasks, implying that the phenomena described in §3 encode stronger word relationships in target space. We show how word2vec target vectors are advantaged in word-search style tasks like analogies and cross-lingual alignment, while context vectors have the advantage in document-level tasks like STS and IR.

Because context vectors do not attract and repel target vectors with equal frequency (see §3), there is higher variability in their length; vectors likely become large to produce high positive dot-products with neighbors or low negative dot-products with non-neighbors. Figure 4 shows that the norms of context vectors are an order of magnitude larger than those of target vectors. This justifies the use of weighted sums. An unweighted addition of small target and large context vectors results in a set that resembles context space. An 80:20-sum may be closer to the ideal of an even sum.

Further analysis suggests that context vectors are ill-equipped for some semantic tasks. Table 4 shows that context space has higher inertia in k-means clustering, indicating that it is harder to cluster into meaningful semantic groups.

Further statistics gathered from word2vec target and context spaces are surprising. (See Table 4.) The extremely small mean, minimum, and maximum cosine distances from the centroid vector and the small standard deviation imply that context vectors are clustered tightly in cosine distance around a centroid. The high skewness and extremely high kurtosis indicate existence of extreme outliers. These properties increase the likelihood of selecting an incorrect vector in tasks that search the vocabulary space, such as analogies and cross-lingual alignment (where context vectors perform worst). Target vectors show none of these disadvantages.

In contrast, although context vectors perform worse on these word-search tasks, they are well-suited to tasks in which word vectors within a sentence or document are summed to form bag-of-words representations, such as STS or IR. Context vectors' more variable norms play to their advantage here, making them preferable to target vectors. Table 5 shows how context vectors for stop words (defined by SpaCy model

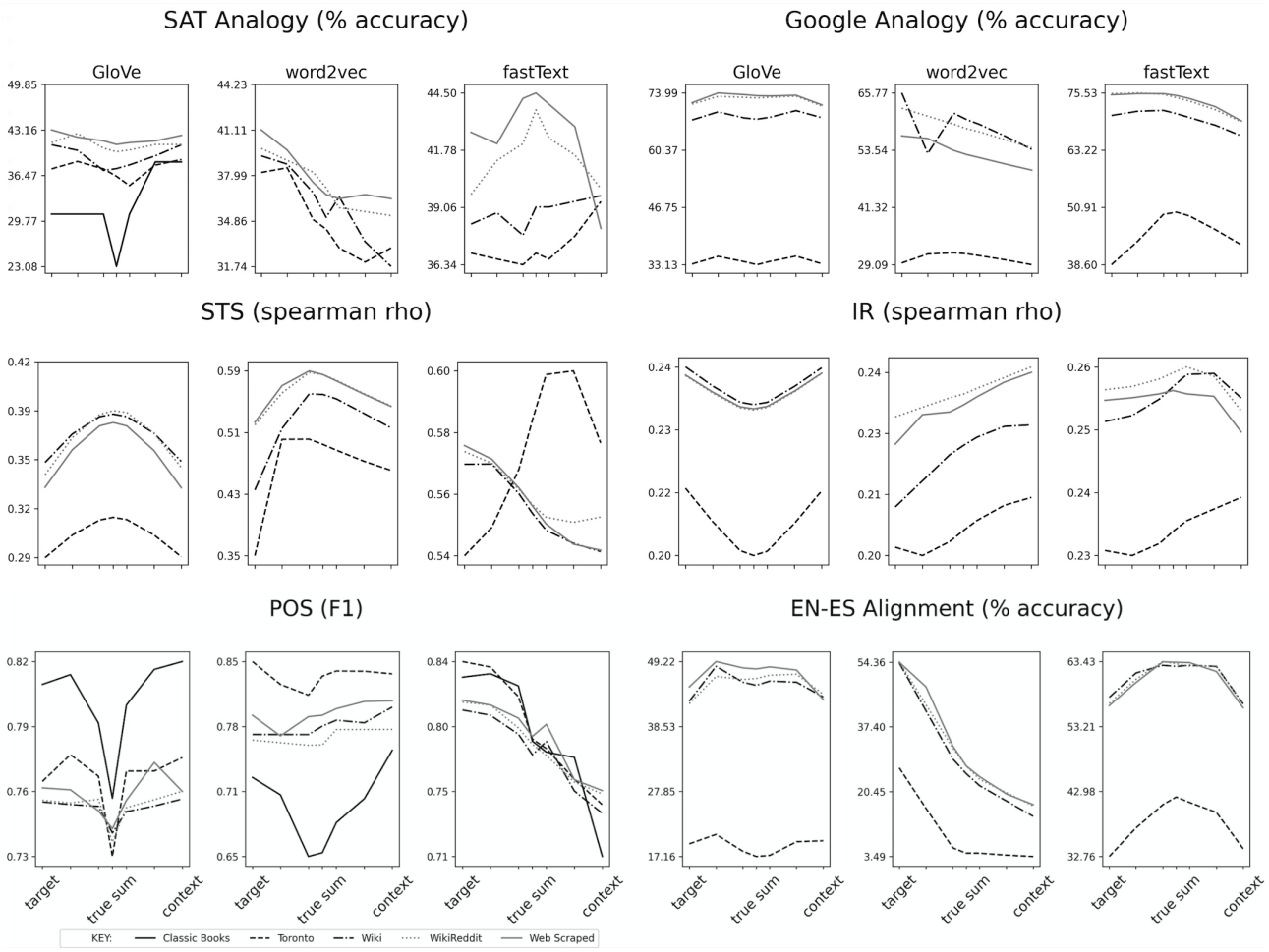


Figure 2: Algorithm performance grouped by task. X-axis ticks correspond left to right to (target, context) weightings of (1.0,0.0), (0.8,0.2), (0.6,0.4), (0.5,0.5), (0.4,0.6), (0.2,0.8), and (0.0,1.0). Results for the smallest corpus are omitted when they are so poor that they impair the visible contrast between other scores.

	SAT Ana.	Google Ana.	STS	IR	POS	EN-ES
word2vec	0%	0%	12.5%	5.27%	0%	0%
fastText	4.39%	0.2%	3.90%	1.52%	0%	9.72%
GloVe	5.22%	0.9%	0%	3.00%	8.8%	2.52%

Table 3: Percentage improvement by target/context weight tuning, over default target/context weighting, for each embedding algorithm and task shown in Figure 2. Improvement of 0% indicates that the default weighting performed best.

`en_core_web_sm`) are smaller than average. This equips context vectors for tasks involving sums of word vectors. It means that vectors carrying less semantic information will be play a less significant role in bag-of-words sentence representations, which will then be less noisy and more closely resemble the vectors for their meaningful keywords.

6.2 fastText Analysis

The fastText algorithm constructs vectors in a similar way to word2vec. However, fastText vectors display different performance patterns from word2vec across tasks. Recall from §3 that fastText target vectors (and not context) benefit from sub-word information. This seems to play a large role in their performance. Sub-word information is useful in POS-tagging, where English morphology can indicate part of speech, and Google analogies, which involve both derivational and

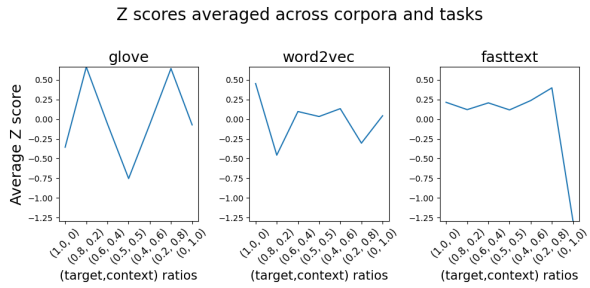


Figure 3: Average Z-scores (standard deviations from the mean) across all training corpora and all tasks in Figure 2.

	w2v tgt	w2v ctx	GloVe tgt	GloVe sum	fT tgt	fT ctx
mean	0.74	0.03	0.76	0.73	0.53	0.29
std. dev.	0.18	0.03	0.20	0.23	0.06	0.13
skewness	-0.04	7.63	-0.18	-0.12	0.22	0.75
kurtosis	-0.31	1e+2	0.06	-0.18	0.15	0.61
min.	0.16	3e-3	0.06	0.04	0.21	2e-3
max.	1.36	0.96	1.90	1.89	0.87	1.26
mode.	1.15	0.46	1.50	1.67	0.56	1.10
inertia (1e6)	1.13	5.31	3.73	11.0	24.8	28.3

Table 4: **Statistics on cosine distances from each vector to centroid** for embeddings trained on Wikipedia, and average inertia for k-means clustering over 6 values of k ($k \in \{5, 10, 15, 20, 25, 30\}$).

	w2v tgt	w2v ctx
Stop word avg. norm	2.70	7.07
Content word avg. norm	2.22	19.66
Stop norm as % of content norm	122%	36.0%

Table 5: In context space (and not target), stop words have smaller norms than other words.

inflectional morphological processes. It also appears useful for semantic textual similarity whenever large training corpora are used.

Interestingly, context vectors perform best in the IR test, which involves summing the word vectors in long documents to form (bag-of-words) document representations $\bar{D}$. Since fastText target vectors are already bag-of-subwords representations (as noted in §2), the treatment of documents as large bags of unordered subwords may dilute the usefulness of the representation.

A particularly notable trend is performance on the cross-lingual alignment task. As expected from its widespread application to multilingual settings, fastText outperforms GloVe and word2vec. But in contrast to the conventional use of target vectors, target-context sums are the best-performing combinations. (In §7.1 we apply these vectors to a similar application and find 80:20 sums to be the best combination.) We hypothesize that this is because of morphological differences between Spanish and English. Sub-word information is useful for interpreting meaning in both languages, but over-dependence on these characteristics may cause failures due to distributional differences in morphology. Analysis of these trends across more language pairs is a topic for future research.

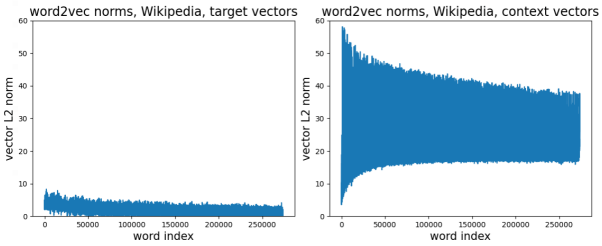


Figure 4: Norms of word2vec target and context vectors trained on Wikipedia, ordered from most common tokens to least

6.3 GloVe Analysis

As discussed in §3, GloVe target and context space are structurally similar. As a result, GloVe performance graphs in Figures 2 and 3 are mostly symmetrical. This leaves the question of why sums perform well on some tasks and poorly on others.

Recall from §4.2 that the Google Analogy Test is composed of nine sub-tests of grammatical analogies and five of semantic analogies. We analyzed vector performance on a fine-grained breakdown of Google Analogy subsets and found that in individual sub-tests, performance varies in a regular way: sums perform well on semantic analogies and poorly on grammatical analogies. Results from two sub-tests are in Figure 5. It appears GloVe's true sum vectors (its default) configured to these semantic questions when the algorithm was tuned on Google analogies, perhaps because the two largest sub-tests in the test set are capital-country and city-state relations. Observe the high accuracy achieved by GloVe default on the semantic task in Figure 5 (90–98%). Trends suggest, however, that the default sum performs worse more generally: In grammatical Google analogy sub-tests, SAT analogies, IR, and POS. Analysis backs this finding: True sum space has higher inertia in k-means clustering than target space (see Table 4), suggesting it is more difficult to cluster meaningfully, and it is the least robust GloVe combination (see Figure 3.)

6.4 Corpus Effect

Performance trends in our experiments are generally consistent across training corpora. Because of the few

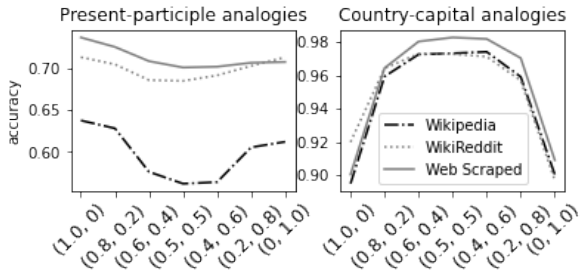


Figure 5: Example GloVe results across embedding combinations for grammatical (left) and semantic (right) analogies

exceptions to this generality and our primary focus on target/context weighting, we did not conduct an in-depth analysis of their causes. Here we note some observations briefly.

For fastText on SAT Analogies, Web Scraped and WikiReddit exhibit different trends from the other corpora. As Table 2 shows, these two corpora yield larger vocabularies than the others. Since SAT analogies rely on embeddings for less frequent vocabulary words, this could be a cause for the trend difference.

We see an aberration from general trends in the performance of fastText vectors trained with Toronto Books on STS. The STS test consists of many pairs of photo captions. The Toronto Books Corpus consists of 61.2% fiction books (Bandy and Vincent, 2021), which is likely rich in descriptive language. The STS test suite also contains answer comparisons and news captions. Dialogue in fiction books could be beneficial for detecting answer similarity. And non-fiction books may be particularly helpful for news caption comparisons. It is difficult to discern why these benefits would be manifest in the case of fastText and with an aberrant target/context trend. Drawing more conclusions may require a more in-depth study into corpus effect on these summed embeddings.

Note that in some experiments, the weighting of target and context determines whether performance from the Classic Books corpus can be comparable to the other corpora. This suggests an opportunity for target/context weight tuning to improve results in very low-resource settings. We discuss this more in §7.4.

7 Applications

Based on our findings, we recommend that practitioners tune the target/context embedding weight for downstream applications. The following two examples demonstrate that by doing so, it is possible to improve upon default vector performance using applications from published literature.

Note that our baselines differ from published results, which employed downloadable pre-trained embeddings that were extensively optimized. This approach yields high-quality results, but it does not allow for comparison of target/context combinations since only one combination is typically released online. We therefore used our own embeddings, trained on the Wikipedia corpus, to explore relative performance across a spectrum of target/context combinations. We encourage future practitioners to release both embedding sets.

7.1 MUSE Cross-lingual Alignment

Cross lingual alignment is a popular approach to multilingual text representations. Algorithms learn a transformation matrix to map embeddings from one language into another. Lample et al. (2018) accomplish this via an unsupervised adversarial algorithm, MUSE.

Using Lample et al.'s methods, we present results for different target/context combinations of fastText vectors. We selected fastText for this experiment to match Lample et al.'s (2018) implementation, and additionally because fastText performed the best on supervised cross-lingual alignment in §5 (the true sum combination, to be precise). To score in this unsupervised case, we query a fixed number of source word embeddings and measure accuracy for correct target retrieval for $k = 1, 5, 10$ nearest neighbors. In this task, we found that the 80:20 sum outperformed all other combinations. See Table 6. Interestingly 20:80 sum and context vectors perform significantly worse than the other combinations we tested, suggesting that the absence of the sub-word enriched target embeddings leads to degradation of performance.

7.2 Harvesting Common-sense Navigational Knowledge for Robotics

In this section we present an additional recommendation to practitioners. When optimizing for target/context weight via differentiation, take care with the choice of objective function, and consider the complexity of high-dimensional vector spaces.

Fulda et al. (2017b) used a novel distance metric to extract navigational relationships between objects for robotics applications, the Directional Scoring Method (DSM). They evaluated this method on a series of ground-truth object relations, contained in the BYU Analogical Reasoning Dataset. (See https://github.com/NancyFulda/BYU-Analogical-Reasoning-Dataset.) Using word2vec skip-gram vectors as the original authors did, we ventured to find the optimal target/context weight $\lambda \in [0, 1]$ for this application (where for each word w, the embedding vector $v_w = \lambda t_w + (\lambda - 1)c_w$). We grid

	default	*80:20 sum*	*60:40 sum*	*50:50 sum*	*40:60 sum*	*20:80 sum*	*context*
k=1	64.14%	**65.70%**	65.55%	63.69%	62.49%	58.02%	51.34%
k=5	81.02%	**81.83%**	81.75%	80.52%	79.56%	75.94%	69.56%
k=10	84.91%	**85.60%**	85.50%	84.57%	83.86%	80.72%	75.25%

Table 6: MUSE alignment accuracy percentages with fastText English and Spanish vectors. Best results are **bold**. The 80:20 combination outperforms all others.

searched over eleven λ values to maximize accuracy ($\lambda \in \{0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0\}$). The default target vectors were quite effective for this application. $\lambda = 1.0$ scored highest for 6 of the 11 analogy categories (accessing containers, affordance, causation, containers, locations for objects, rooms for containers). Results for the other 5 categories are in Table 7.

	belong	rooms objects	tools	trash treas.	travel
default	6.21%	21.1%	3.57%	37.0%	17.6%
λ *GS*	9.66%	21.5%	6.90%	52.2%	18.4%
$\lambda =$	0.9	0.8	0.8	0.4	0.9

Table 7: DSM scores where λ grid search (GS) improved performance. *Default* indicates $\lambda = 1$.

We found that optimizing λ via differentiation was not effective. We constructed an objective, the sum of directional scores for each of the ground-truth analogy answers in a training set, 60% of analogical questions, and maximized the objective over λ. Because this objective minimized DSM-distance between ground truth vectors and DSM-hypothesis vectors, the optimal value of λ defaulted to the most tightly clustered vector space available. In our case with word2vec, $\lambda = 0$ was selected every time, since word2vec context vectors are tightly clustered in DSM space. This did not maximize the probability of choosing the correct vector.

Modifying the objective function by subtracting directional scores for all incorrect answers circumvented this issue and allowed for more diverse selections of optimal λ, but it still was not effective. We found that the value of λ that maximizes DSM scores for all correct answers and minimizes scores for all incorrect answers does not necessarily result in higher answer accuracy, regardless of whether we used a subset of questions for training or the whole set. This is likely due to intractable subtleties in high-dimensional vector geometries, a phenomenon for further investigation.

The results from this application do not preclude the use of differentiation or other meta-learning techniques to tune the target/context weight λ for other tasks. We strongly encourage such investigations. But until further conclusions, we urge that objectives to optimize λ be attempted with caution and thorough verification.

7.3 Summary Recommendations

Our observations suggest that values in Table 8 may be the most reasonable choices of target/context combinations for each embedding algorithm and task. Because even similar tasks may differ in nature, however, we encourage all practitioners to optimize target/context weights via grid search whenever possible.

While our recommendation is to tune/optimize, we recognize that many researchers, especially those applying word embeddings in research areas outside of computational linguistics, may not have the resources to tune their own weight parameters. We therefore provide recommendations on which simple linear combinations are broadly applicable to various task types.

7.4 Potential for Low-resource and Multilingual NLP

Our results suggest that tuning target/context weight can, in some cases, elevate the performance of low-resource embeddings to the level of higher-resource systems. One of the most promising areas for improvement of static word representations is NLP for low-resource languages. Many low-resource languages do not have high-quality contextual embedding tools such as BERT (Devlin et al., 2019) and lack the resources to train data-hungry BERT-like models. Many of these languages rely on improvements in static embedding technologies for accurate representations in NLP.

Multiple of the word2vec, GloVe, and fastText applications listed in §1 and §2 are for low-resource domains. Among the works referenced in this paper alone, we find examples of static embeddings applied to technologies for Amharic, Azerbaijani, Belarusian, Bengali, Galician, Gujarati, Hausa, Marathi, Punjabi, Somali, Tamil, Telugu, Uighur, Urdu, Uzbek, Yoruba, and more (Qi et al., 2018; Pourdamghani et al., 2018; Khalid et al., 2021; Akhtar et al., 2017).

A major limitation of our study is that, given the large number of independent variables we tested already, we were constrained to applications involving English (and some with Spanish, another high-resource language). Unfortunately, this limitation inhibits us

	word2vec	fastText	GloVe
word search tasks	target	**target-heavy/true sums**	target/context
sentence textual similarity	**true sum**	target/20:80 sum	true sum
bag-of-words representations	context	**context-heavy sums**	target/context
semantic Google analogies	target	target-heavy/true sums	**true sum**
grammatical analogies	target	**target-heavy/true sums**	target/context
other analogies	target	**target-heavy/true sums**	target/context
cross-lingual alignment	target	**80:20 sum/true sum**	80:20 sum
POS tagging	**target/context**	target	target-/context-heavy
overall	target	**target/20:80 sum**	80:20 sum/20:80 sum

Table 8: Recommended target/context embedding usage by task and embedding algorithm. The algorithm that performed best on each task type in our experiments is **bold**. Practitioners are cautioned that even similar tasks may differ in nature, and that the general trends indicated here may not hold in all use cases.

from drawing concrete conclusions about performance trends and their dependence on training language. This is a primary area of potential for future research. We hope to see more targeted studies addressing the effectiveness of target/context weight tuning on low-resource tasks, particularly for fastText vectors, which are often used in multilingual settings. Since fastText vectors formed by target-context sums combine morpheme information with full word information, they could be valuable in applications for morphologically rich languages, such as Arabic, Finnish, and Quechua.

8 Conclusion

By leveraging unconventional combinations of target and context vectors learned by GloVe, fastTest, and word2vec, we achieve improvements of up to 12.5% on common word embedding tasks such as POS-tagging and IR, thus elevating the usefulness of these popular and inexpensive word representations for NLP tasks. Experiments with 126 embedding sets on six generic tasks and two downstream applications show that tuning the hyperparameter of target and context weight for downstream tasks can improve performance significantly over default settings, increasing accuracy by 0.69% to 1.56% on MUSE cross-lingual alignment and by up to 15.2% on navigational robotics benchmarks.

Analysis suggests that target-heavy word2vec combinations are most suited to tasks involving single-word relationships, while context vector information is useful in summed sentence representations. We further observe that GloVe default settings perform best on tasks for which GloVe was tuned but tend to perform poorly on others, and that fastText target vectors excel in tasks such as POS-tagging, where sub-word information is particularly relevant. These findings reveal a disconnect between the maximum potential of static embedding algorithms and the ways in which they are typically used. In a majority of cases, the performance of pre-trained word embeddings could be improved by tuning the target/context weight hyperparameter. Furthermore, because a target/context weighting is typically chosen prior to the release of extensively pre-trained word vectors, the possibility of exploring various target/context weightings has typically not been made available to subsequent researchers. In alignment with our results, we urge those who design and train static word embedding models to release both target and context vector sets.

The software and embeddings used in our experiments will be released publicly under the MIT license. Given the widespread use of GloVe, word2vec, fastText, and other static embeddings, there is a need for deeper understanding of target and context interactions. Directions of future work in this area include the semantic content contained in context and target embeddings; the interplay between embedding algorithm, corpus size, and corpus genre; vector normalization methods to avoid norm imbalance; distance metrics not based on cosine similarity; and paired-embedding algorithms where context and target spaces are used individually.

References

Akhtar, Syed Sarfaraz, Arihant Gupta, Avijit Vajpayee, Arjit Srivastava, and Manish Shrivastava. 2017. Word similarity datasets for Indian languages: Annotation and baseline systems. In *Proceedings of the 11th Linguistic Annotation Workshop*, pages 91–94, Valencia, Spain. Association for Computational Linguistics.

Andrus, Berkeley and Nancy Fulda. 2020. Immersive gameplay via improved natural language understanding. In *Foundations of Digital Games 2020*.

Attardi, Giusepppe. 2015. Wikiextractor. https://github.com/attardi/wikiextractor.

Bandy, John and Nicholas Vincent. 2021. Addressing "documentation debt" in machine learning: A retrospective datasheet for bookcorpus. In *Proceedings*

of the Neural Information Processing Systems Track on Datasets and Benchmarks, volume 1.

Bender, Emily M., Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 610–623, New York, NY, USA. Association for Computing Machinery.

Bojanowski, Piotr, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Brown, Zachary, Nathaniel Robinson, David Wingate, and Nancy Fulda. 2020. Towards neural programming interfaces. *Advances in Neural Information Processing Systems*, 33:17416–17428.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Dufter, Philipp, Nora Kassner, and Hinrich Schütze. 2021. Static embeddings as efficient knowledge bases? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2353–2363.

Fulda, Nancy, Daniel Ricks, Ben Murdoch, and David Wingate. 2017a. What can you do with a rock? affordance extraction via word embeddings. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 1039–1045.

Fulda, Nancy and Nathaniel Robinson. 2021. Improved word representations via summed target and context embeddings. In *2020 IEEE 18th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*.

Fulda, Nancy, Nathan Tibbetts, Zachary Brown, and David Wingate. 2017b. Harvesting common-sense navigational knowledge for robotics from uncurated text corpora. In *Proceedings of the First Conference on Robot Learning*.

Gupta, Piyush, Inika Roy, Gunnika Batra, and Arun Kumar Dubey. 2021. Decoding emotions in text using glove embeddings. In *2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, pages 36–40.

Hamilton, William L., Jure Leskovec, and Dan Jurafsky. 2016. Diachronic word embeddings reveal statistical laws of semantic change. *CoRR*, abs/1605.09096.

Jain, Vaibhav. 2020. GloVeInit at SemEval-2020 task 1: Using GloVe vector initialization for unsupervised lexical semantic change detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 208–213, Barcelona (online). International Committee for Computational Linguistics.

Jansen, Stefan. 2017. Word and phrase translation with word2vec. *CoRR*, abs/1705.03127.

Khalid, Usama, Aizaz Hussain, Muhammad Umair Arshad, Waseem Shahzad, and Mirza Omer Beg. 2021. Co-occurrences using fasttext embeddings for word similarity tasks in urdu. *CoRR*, abs/2102.10957.

Khatri, Akshay and Pranav P. 2020. Sarcasm detection in tweets with BERT and GloVe embeddings. In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 56–60, Online. Association for Computational Linguistics.

Khatua, Aparup, Apalak Khatua, and Erik Cambria. 2019. A tale of two epidemics: Contextual word2vec for classifying twitter streams during outbreaks. *Information Processing & Management*, 56(1):247–257.

Kingma, Diederik P and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR (Poster)*.

Kocmi, Tom and Ondřej Bojar. 2017. An exploration of word embedding initialization in deep-learning tasks. In *Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017)*, pages 56–64, Kolkata, India. NLP Association of India.

Lahiri, Shibamouli. 2014. Complexity of Word Collocation Networks: A Preliminary Structural Analysis. In *Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 96–105, Gothenburg, Sweden. Association for Computational Linguistics.

Lample, Guillaume, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *International Conference on Learning Representations*.

Lewis, Mike, 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.

Lin, Zhou, Qifeng Zhou, and Langcai Cao. 2021. Two-stage encoder for pointer-generator network with pretrained embeddings. In *2021 16th International Conference on Computer Science & Education (ICCSE)*, pages 524–529. IEEE.

Liu, Nelson F., Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019a. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.

Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *NIPS*, pages 3111–3119. Curran Associates, Inc.

Mikolov, Tomas, Wen-tau Yih, and Geoffrey Zweig. 2013c. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, Georgia. Association for Computational Linguistics.

Minaee, Shervin, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. 2021. Deep learning–based text classification: A comprehensive review. *ACM Comput. Surv.*, 54(3).

Mitra, Bhaskar, Eric Nalisnick, Nick Craswell, and Rich Caruana. 2016. A dual embedding space model for document ranking. This paper is an extended evaluation and analysis of the model proposed in a poster to appear in WWW'16, April 11 - 15, 2016, Montreal, Canada.

Nalisnick, Eric, Mitra Bhaskar, Nick Craswell, and Rich Caruana. 2016. Improving document ranking with dual word embeddings. In *WWW '16 Companion: Proceedings of the 25th International Conference Companion on World Wide Web*, pages 83–84.

Ni, Chien-Chun. 2015. Multiple choice question (MCQ) dataset. https://www3.cs.stonybrook.edu/ chni/post/mcq-dataset/.

Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543.

Peters, Matthew, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. 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 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Peterson, Joshua C. 2019. OpenWebText. https://github.com/jcpeterson/openwebtext.

Pickard, Thomas. 2020. Comparing word2vec and GloVe for automatic measurement of MWE compositionality. In *Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons*, pages 95–100, online. Association for Computational Linguistics.

Pourdamghani, Nima, Marjan Ghazvininejad, and Kevin Knight. 2018. Using word vectors to improve word alignments for low resource machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 524–528, New Orleans, Louisiana. Association for Computational Linguistics.

Premjith, B. 2019. Part of speech tagging machine learning deep learning word2vec fastText. https://github.com/premjithb/Part-of-Speech-Tagging-Machine-Learning-Deep-Learning-Word2vec-fasttext.

Qi, Ye, Devendra Sachan, Matthieu Felix, Sarguna Padmanabhan, and Graham Neubig. 2018. When and why are pre-trained word embeddings useful for neural machine translation? In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages

529–535, New Orleans, Louisiana. Association for Computational Linguistics.

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Robinson, Nathaniel, Zachary Brown, Timothy Sitze, and Nancy Fulda. 2021. Text classifications learned from language model hidden layers. In *2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, pages 000207–000210. IEEE.

Sabet, Masoud Jalili, Philipp Dufter, and Hinrich Schütze. 2020. Simalign: High quality word alignments without parallel training data using static and contextualized embeddings. *CoRR*, abs/2004.08728.

Speer, Robyn, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 4444–4451. AAAI Press.

Strubell, Emma, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650, Florence, Italy. Association for Computational Linguistics.

Turney, Peter D. 2006. Similarity of semantic relations. *Computational Linguistics*, 32(3):379–416.

Wilson, Theresa, Zornitsa Kozareva, Preslav Nakov, Alan Ritter, Sara Rosenthal, and Veselin Stoyanov. 2013. Sentiment analysis in twitter. http://www.cs.york.ac.uk/semeval-2013/task2/.

Yuan, Weizhe, Graham Neubig, and Pengfei Liu. 2021. BARTScore: Evaluating generated text as text generation. *Advances in Neural Information Processing Systems*, 34:27263–27277.

Zhang, Tianyi, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

Zhu, Yudong, Di Zhou, Jinghui Xiao, Xin Jiang, Xiao Chen, and Qun Liu. 2020. HyperText: Endowing FastText with hyperbolic geometry. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1166–1171, Online. Association for Computational Linguistics.

Zhu, Yukun, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *CoRR*, abs/1506.06724.

Building Analyses from Syntactic Inference in Local Languages: An HPSG Grammar Inference System

Kristen Howell, University of Washington and LivePerson Inc., USA kphowell@uw.edu
Emily M. Bender, University of Washington, USA ebender@uw.edu

Abstract We present a grammar inference system that leverages linguistic knowledge recorded in the form of annotations in interlinear glossed text (IGT) and in a meta-grammar engineering system (the LinGO Grammar Matrix customization system) to automatically produce machine-readable HPSG grammars. Building on prior work to handle the inference of lexical classes, stems, affixes and position classes, and preliminary work on inferring case systems and word order, we introduce an integrated grammar inference system called BASIL that covers a wide range of fundamental linguistic phenomena. System development was guided by 27 genealogically and geographically diverse languages, and we test the system's cross-linguistic generalizability on an additional 5 held-out languages, using datasets provided by field linguists. Our system out-performs three baseline systems in increasing coverage while limiting ambiguity and producing richer semantic representations, while also producing richer representations than previous work in grammar inference.

1 Introduction

Machine-readable grammars for human languages that are grounded in theoretical syntactic formalisms can be useful tools in the context of endangered language documentation and revitalization. First, they support treebanking (Oepen et al., 2002), which in turn supports data exploration (Letcher and Baldwin, 2013; Bouma et al., 2015); and second, they facilitate the development of tools such as grammar checkers (da Costa et al., 2016) and automated tutors (Hellan et al., 2013). In spite of these advantages, the use of such grammars is hindered by the time-consuming process of developing them together with the need of a specific skillset required for grammar engineering, which is distinct from the skills involved in documentation itself. We are therefore motivated to investigate whether we can create machine-readable grammars automatically.[1] Endangered languages represent scenarios where the type of resources required for typical natural language processing techniques are scarce to non-existent. Furthermore, the output we are targeting goes well beyond simple labels or even structured representations, but rather must be a coherent and well-formed formal object — a grammar.

Fortunately, we have two rich sources of linguistic knowledge from which to work: The first is corpora of interlinear glossed text (IGT), annotated by field linguists during the process of documentation and analysis. Due to the efforts of field linguists and archivists, a number of archives (many of which we list in Appendix A) make IGT data publicly available. An example from Chintang [ISO 639-3: ctn] is shown in (1). Such annotations are linguistically rich, showing what grammatical information is marked morphologically and providing further information implicitly via a translation into a language of broader communication (in all examples we work with, this language of broader communication is English). Using the methodology of annotation projection, as applied to IGT (Xia and Lewis, 2007; Georgi, 2016), we can leverage parsers available for the translation language and project structural information such as part-of-speech (POS) tags and syntactic dependencies onto words in the target language.

(1) Aru unisokoniŋ.
aru u-ŋis-u-kV-niŋ
another 3nss/A-know-3P-IND.NPST-NEG
'They did not know another [language].' [ctn]
(Bickel et al., 2013a)

The second source of linguistic knowledge that we have in hand is the LinGO Grammar Matrix customization system (Bender et al., 2002, 2010; Zamaraeva et al., forthcoming), which maps from relatively

[1] This is similar in spirit to the work of Sarveswaran et al. (2019) who present an effort to create FSMs to provide computational benefits in the context of morphological analysis without requiring additional technical skillsets.

simple grammar specifications to full-fledged machine-readable grammars, couched in the framework of Head-driven Phrase Structure Grammar (HPSG; Pollard and Sag 1994; Müller et al. 2021), and compatible with DELPH-IN[2] processing tools. The Grammar Matrix customization system consists of a core grammar, hypothesized to be shared across languages, and a series of typologically-informed libraries of analyses of cross-linguistically variable phenomena.

Leveraging these sources, the question we investigate here is whether and how we can create machine-readable HPSG grammars for typologically diverse local[3] and/or endangered languages on the basis of corpora of IGT and the Grammar Matrix. In particular, we build on the open-source code base provided by the AGGREGATION project (Bender et al., 2014, inter alia) to produce the following contributions: (1) We integrate all existing inference modules into a single system to which (2) we add modules for additional grammatical phenomena and (3) where previous end-to-end testing treated only a single language, we use 27 diverse languages in development, doing end-to-end system testing on 9 of the 27, and then evaluate on 5 additional held-out languages not considered during system development.

We begin by situating our work on grammar inference against the broader background of automatic grammar generation in Section 2 and then provide background on the AGGREGATION project in Section 3. Section 4 describes our methodology for grammar inference, including lexical, morphological and syntactic aspects of an inferred grammar. In Section 5, we describe the languages we used in system development and how we use the DELPH-IN suite of software tools to evaluate the grammars we create by parsing and treebanking held-out data from each language. We use that same methodology for held-out languages to evaluate the generalizability of the system, finding that though the coverage of the grammars is still limited, the proposed methodology generally produces higher quality grammars than three baseline approaches. The languages we test on and the results of this evaluation are presented in Sections 6 and 7. Finally, Section 8 provides error analysis and discussion. We conclude in Section 9 with discussion of applications of grammars produced in this fashion.

[2] www.delph-in.net

[3] These are often called 'low-resource languages', but Bird (2022) argues that this label projects a number of Eurocentric beliefs onto these languages. Bird proposes describing languages as *standardized*, *local* and *contact* rather than *high* and *low resource*.

2 Automatic Grammar Generation

Interest in creating machine-readable grammars is likely as old as the field of computational linguistics itself, with published work in *grammar engineering* — the process of creating machine-readable grammars by hand — going back at least as far as Zwicky et al. (1965) and continuing into the present day. Our work in grammar inference builds on grammar engineering work (in the form of the Grammar Matrix; Bender et al., 2002, 2010; Zamaraeva et al., forthcoming), but also fits into a tradition of work on *automatic grammar generation*, which is the development of systems that automatically create grammars on the basis of data. Within automatic grammar generation, we distinguish four broad categories of approaches, differentiated by the types of inputs they take: *grammar induction from strings* — automatic grammar generation based on text alone (§2.1); *grammar extraction* — automatic grammar generation based on treebanks (§2.2); *grammar induction from meaning representations* — automatic grammar generation based on strings paired with some form of semantic representation (§2.3); and *grammar inference* — automatic grammar generation based on text annotated with partial grammatical information but not full parse trees or logical forms (§2.4).

Just as these four approaches to grammar generation differ in their input, they also differ in the types of grammars they can produce. Grammar induction, if working from strings alone, will produce noisy representations that align only partially with structures created by linguists. Grammar extraction will produce grammars that provide the same kind of representations as given in the source treebank and similarly, grammar induction based on strings paired with semantic representations will produce grammars that can output those semantic representations. In each of these cases, the generated grammar will also typically include a parse selection model, based on observed patterns in the corpus. Grammar inference systems, by contrast, draw on both partial annotation in their input data and some external source of grammatical knowledge. For this reason, the inferred grammars can generate richer representations than those found in the input.

2.1 Grammar Induction from Strings

Often characterized as an incomplete data problem (see inter alia Klein and Manning, 2001), where the complete data would be a corpus of trees, *grammar induction* from surface strings seeks to produce grammars solely on the basis of text. Early grammar induction work focused on producing context-free grammars (CFGs), which involved two components: (1) identifying con-

stituents and (2) identifying their categories (see Klein and Manning, 2001, 2002). Klein and Manning (2004) improved upon this work by inducing an unlabeled syntactic dependency grammar and combining it with the induced CFG for better performance parsing over English [eng], German [deu] and Mandarin [cmn]. This basic approach has informed work which further tuned the algorithm by preferring short vs. long dependencies and testing on additional languages, as in Smith and Eisner 2006. One shortcoming of these approaches is that they only take into account contiguous dependencies. Bod (2009) introduces an approach that allows discontiguous subtrees and thereby handles non-adjacent dependencies. Most recently, neural nets, such as BERT (Devlin et al., 2019), have proven effective in producing unlabeled dependency parses, as demonstrated by Hewitt and Manning (2019), although only parses and not a human-interpretable grammar have been generated. While unlabeled syntactic dependencies can be inferred from text and are useful for some tasks, they do not provide any information regarding the type of syntactic relationship between two constituents. Therefore, other methodologies of automatic grammar generation have focused on using inputs that are encoded with more linguistic information.

Still another strand of recent work seeks to improve grammar induction by using strings (still without linguistic labels) that are captions of still images (Shi et al., 2019; Zhao and Titov, 2020) or descriptions of videos (Zhang et al., 2021). These sources of grounding have been shown to improve recall of different constituent types, but the resulting parsers still produce quite impoverished and noisy representations.

2.2 Grammar Extraction

In contrast with the impoverished input used by grammar induction from surface strings, grammar extraction uses the syntactic information available in treebanks — collections of syntactic trees — to define grammars. Typically these grammars are produced by walking the trees in a treebank, collecting rules that could produce those structures and pruning to remove redundant rules (Krotov et al., 1998).

Because an extracted grammar is informed by the formalism and theory implicit in the tree structures in the input, it will produce trees with roughly the same amount of syntactic information as the formalism used to create the treebank. This can range from context-free grammars (CFG), as in Krotov et al. 1994, to grammar formalisms such as HPSG, as in Simov 2002. However, while the level of detail in the treebanked parses limits that of the resulting grammar, work has been done to extract a grammar in a different formalism than that represented in the input. Xia (1999), for example, proposed an algorithm to do additional bracketing on the Penn Treebank II-style trees (Marcus et al., 1994) in order to extract a Lexical Tree Adjoining Grammar (LTAG), which was more expressive than the CFG in the input. Similarly, Hockenmaier and Steedman (2007) present an approach to converting the Penn Treebank to Combinatory Categorial Grammar (CCG) representations, adding significant information, from which CCG grammars can then be extracted (e.g. Hockenmaier and Steedman, 2002; Clark and Curran, 2004). Neural networks have also been used to generate parse trees based on syntax trees in the training data. KERMIT (Zanzotto et al., 2020) generates syntactic parses of the same form as those in the training data and lends a great deal of interpretability to the underlying BERT (Devlin et al., 2019) model, although it does not produce a grammar or human-interpretable rules.

In principle, grammar extraction is possible for any language for which there is a treebank and recent work has leveraged the Universal Dependencies Treebank (Nivre et al., 2016), a collection of dependency treebanks for over 100 languages, to generate grammars for a wide range of languages (see inter alia Agić et al., 2016; Noji et al., 2016; Han et al., 2019). Our goals in this work, however, are to generate grammars for local languages,[4] many of which are not represented in the UD collection, and to produce syntactic and semantic representations which are richer than dependency parses.

2.3 Grammar Induction from Meaning Representations

In contrast with grammar extraction which relies on a treebank of syntactic parses, grammar induction from meaning representations relies on *sembanks*, typically pairing sentences with either semantic dependencies or logical forms. The types of semantic representations used in this work have ranged from formal query language (Kate et al., 2005; Kate and Mooney, 2006) to semantic dependencies from the Redwoods treebanks, which are based on Minimal Recursion Semantics (MRS; Copestake et al., 2005) as in Buys and Blunsom 2017 and Chen et al. 2018. The input is not always limited to meaning representations alone, and for example, previous work has also used additional input lexical templates to better handle morphological complexity (Kwiatkowski et al., 2011).

Due to the richness of semantic information in the input, grammars induced from text paired with semantic representations rather than text alone are capable of capturing much more detailed and meaningful semantic relations than the unlabeled syntactic dependency relations produced by grammars induced only from surface forms. Such semantic representations are still, however, constrained by what's available in the

[4] See footnote 3.

training data.

2.4 Grammar Inference

Grammar inference systems take as input a collection of text with partial grammatical annotations and use some external source of grammatical knowledge that is not specific to the language at hand to produce grammars that give richer representations than those produced by grammar induction without requiring a treebank. While these systems generally are not probabilistic and do not necessarily include a parse-selection model, as is common with induced or extracted grammars, they allow us to automatically generate formal linguistic grammars without a treebank.

To produce grammars in the Minimalist Grammar formalism (MG; Stabler, 1996) of the Minimalist Program (Chomsky, 1995), Indurkhya (2020) used a set of sentences annotated for part-of-speech (POS), agreement, predicate-argument structure and clause type (interrogative or declarative). This system inferred a lexicon for English on the basis of those annotations, pruned it with a set of Minimalist axioms, and combined it with a non-language-specific notion of merge (with internal and external subtypes) to create a machine-readable Minimalist Grammar.

Whereas Indurkhya used a custom annotation scheme for the input data, Hellan (2010) and Bender et al. (2014) leveraged the rich annotation already present in interlinear glossed text (IGT), illustrated in (1). IGT is a particularly rich source of data because it includes morpheme segmentation, glosses for each morpheme which encode morpho-syntactic information and a translation into a language with many NLP resources (frequently English). A particularly attractive fact about IGT data is that it is the format broadly used in linguistics to record data during collection and analysis, so IGT corpora exist for many languages that do not otherwise have very much written text.

Hellan (2010) and Hellan and Beermann (2011) inferred grammars using a combination of specially annotated IGT and the grammar engineering toolkit *TypeGram*. TypeGram is based on the DELPH-IN Joint Reference Formalism (Copestake, 2002a) which supports the development of typed feature structure grammars, typically within the HPSG framework. Hellan (2010) positioned TypeGram as a hybrid of HPSG and Lexical Functional Grammar (LFG; Kaplan and Bresnan, 1982). In addition to the annotations of typical IGT, their input data also included labels indicating syntactic properties such as valence patterns and constructions such as passive. The TypeGram resource included grammatical rules which are named by the same inventory of label types and thus could directly instantiate a grammar off of an appropriately annotated corpus. The authors illustrate their system with examples from Ga [gaa] and Kiswahili [swh].

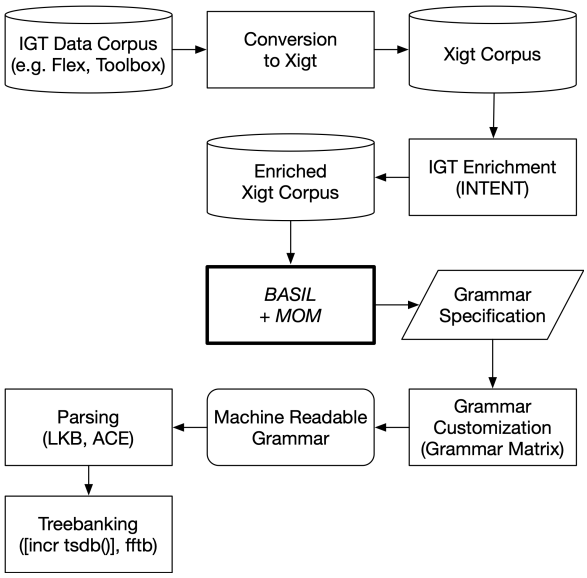


Figure 1: AGGREGATION Pipeline

Bender et al. (2014) also produced HPSG grammars in the DELPH-IN formalism on the basis of IGT data. However, they worked directly from the type of annotations typically produced by documentary linguistics projects, that is, IGT with thorough segmentation and glossing at the morpheme level, but no clause-level annotations. They inferred a lexicon, morphological rules and syntactic properties, and encoded this information in grammar specifications. Using the Grammar Matrix, which allows the user to define a grammar specification that selects from a typologically broad catalog of analyses for different syntactic phenomena and pairs these analyses with a core grammar used across languages, they generated grammars for Chintang [ctn] from their inferred specifications.

Our goal is to create precise syntactic grammars for languages without existing extensive NLP resources, using the rich annotated data that already exists for many of these languages. We build on the approach set forth by Bender et al. (2014), which we describe in detail in the following section. In addition, we extend the typological breadth of work on automatic grammar generation by focusing on languages which are far from the NLP mainstream.

3 The AGGREGATION Project

The AGGREGATION project (Bender et al., 2013, 2014; Howell et al., 2017; Zamaraeva et al., 2017, 2019a), describes its primary goal as providing the benefits of implemented, formal grammars to documentary linguists, without their having to invest time in develop-

ing those grammars by hand. Such grammars are useful for testing linguistic hypotheses against data (Bierwisch, 1963; Müller, 1999; Bender, 2008b; Fokkens, 2014; Müller, 2015) as well as building treebanks which are useful for discovering examples of phenomena in a language (Bender et al., 2012; Letcher and Baldwin, 2013; Bouma et al., 2015). The task of developing a grammar by hand is very time consuming and not likely to be taken up by field linguists already busy with the work of language documentation and description. However, the detailed analysis involved in annotating IGT data (another time consuming task that documentary linguists are doing anyway) provides a very rich starting point for producing these grammars automatically. Therefore, an end-to-end pipeline that begins with an IGT corpus and results in a machine-readable grammar has the potential to serve the language documentation community without requiring additional work on their end, either in the form of data curation or grammar engineering.[5] The AGGREGATION project has produced many key components towards this goal, as well as a rudimentary end-to-end pipeline (tested on Chintang in Bender et al. 2014 and Zamaraeva et al. 2019a). In this work, we build on those components to create a more robust and full-featured pipeline. In this section, we present the overall AGGREGATION pipeline as it is developed in our work, with reference to previous work.

In (2; repeated from 1) we present an example of interlinear glossed text (IGT) from the Chintang Language Research Project (CLRP; Bickel et al., 2013b). Based on the information encoded in this IGT and others in the corpus, our goal is a grammar that parses this sentence to produce an HPSG syntactic representation, like the one in Figure 2, and an MRS semantic representation, as in Figure 3.

(2) Aru unisokoniŋ.
aru u-ŋis-u-kV-niŋ
another 3nss/A-know-3P-IND.NPST-NEG
'They did not know another [language].' [ctn]
(Bickel et al., 2013a)

Inferring an implemented HPSG grammar directly from an IGT corpus would probably be prohibitively difficult, given the intricate nature of the target grammar. However, we have established a pipeline that leverages a number of existing resources to extract information from an IGT corpus and produce a customized grammar for that language. This pipeline, illustrated in Figure 1, expects as its starting point an IGT corpus, typically from Toolbox (SIL International, 2015) or FLEx

[5] Ultimately, we hope to serve the communities whose languages are being documented, whether by outsider or insider linguists, by enabling further language technology. However, the immediate audience for implemented grammars remains linguists as opposed to language teachers and learners.

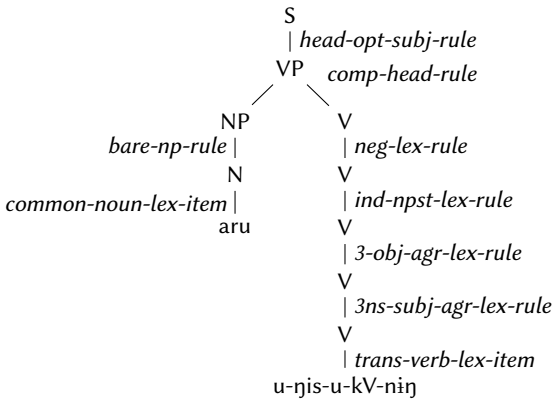
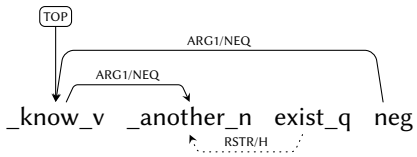


Figure 2: The parse tree for the sentence in (2), which was generated by an inferred grammar of Chintang and corresponds to the semantic representation in Figure 3



Key features on semantic variables:
_know_v (ARG0 {SF *prop*, TENSE *npst*, ASPECT *ind*}, ARG1 {PER *3rd*, NUM *ns*}, ARG2 {PER *3rd*})

Figure 3: A semantic representation for the sentence in (2), generated by an inferred grammar of Chintang

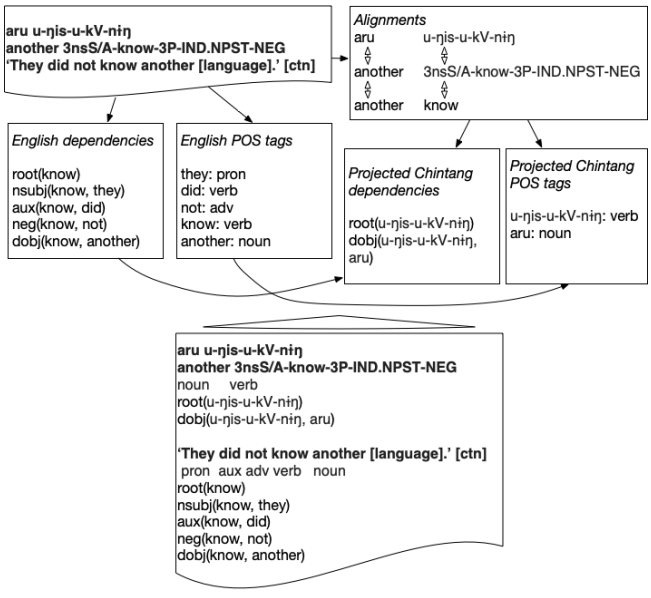


Figure 4: IGT Enriched with INTENT

(also from SIL, see (Rogers, 2010)), that was collected by a field linguist, which we convert to an extensible and flexible XML-based format for IGT data called Xigt (Goodman et al., 2015). We then enrich the IGT using INTENT (Georgi, 2016), which projects syntactic dependencies and part-of-speech (POS) tags onto words in the language from a parse of the English translation, as shown in Figure 4.

The enriched corpus provides four key components that are necessary for grammar inference: morpheme segmentation, glossing, POS tags and syntactic dependencies, which can be seen in the final box in Figure 4. The morpheme segmentation and glossing are provided by the linguist in the source IGT and are necessary to extract a lexicon, infer the morphotactic system and associate morpho-syntactic and morpho-semantic information with the corresponding morphemes. POS tags are often provided in the source IGT, but if they are not, they can be acquired from INTENT. INTENT creates alignments between the English translation and the sentence by leveraging the one-to-one alignment between words of the sentence and words in the gloss line and noisy alignment between the gloss words (frequently English lemmas) and the English translation line. It then parses the English sentence and projects the POS and syntactic dependency tags from the English parse onto the aligned words in the source language. While this approach only provides an approximation, as POS and dependencies do not necessarily map across languages, it serves as a useful starting point for inference. Finally, the projected dependencies allow us to discriminate between arguments, modifiers and conjuncts and to identify different types of constituents in the sentence in order to infer syntactic properties.

Our grammar inference system uses these four components to produce a grammar specification file. As an example of our target output, Figure 5 illustrates some of the values we infer that are relevant to sentential negation in Chintang. Chintang expresses sentential negation with a verbal suffix *-n̈iŋ*. We indicate that negation is expressed with a single morpheme by setting the negation exponence (`neg-exp`) to 1 in the grammar specification. In the morphology section of the grammar specification, we define one or more lexical rules for a morpheme with orthography *n̈iŋ* and morpho-semantic feature `negation: plus`. This grammar specification can be input to the Grammar Matrix customization system (Bender et al., 2002, 2010), which uses stored syntactic analyses to produce customized grammars for languages based on the specification. The customized grammar generated by the Grammar Matrix for this specification will contain the appropriate lexical rule(s) to model negation (Crowgey, 2012), which are illustrated in Figure 6.

```
section=general
  language=Chintang
  iso-code=ctn

section=sentential-negation
  neg-exp=1
  infl-neg=on
  neg-aux=on

section=morphology
  verb-pc14_name=verb-pc14
  verb-pc14_order=suffix
  verb-pc14_inputs=verb-pc1, verb-pc3, ...,
      verb-pc14_lrt1_feat1_name=negation
      verb-pc14_lrt1_feat1_value=plus
      verb-pc14_lrt1_feat1_head=verb
      verb-pc14_lrt1_lri1_inflecting=yes
      verb-pc14_lrt1_lri1_orth=-n̈iŋ
```

Figure 5: A portion of the grammar specification containing (some of) the relevant specifications for sentential negation in Chintang

```
verb-pc5_lrt2-lex-rule := cont-change-only-lex-rule &
                          verb-pc5-lex-rule-super &
  [ C-CONT [ HOOK [ XARG #xarg,
                    LTOP #ltop,
                    INDEX #ind ],
             RELS <! event-relation & [ PRED "neg_rel",
                                        LBL #ltop,
                                        ARG1 #harg ] !>,
             HCONS <! qeq & [ HARG #harg,
                              LARG #larg ] !> ],
    SYNSEM.LKEYS #lkeys,
    DTR.SYNSEM [ LKEYS #lkeys,
                 LOCAL [ CONT.HOOK [ XARG #xarg,
                                     INDEX #ind,
                                     LTOP #larg ],
                         CAT.HEAD verb ] ] ].

verb-pc14_lrt1-suffix :=
%suffix (* -n̈iŋ)
verb-pc14_lrt1-lex-rule.
```

Figure 6: The relevant lexical rule for negation in the Chintang grammar, produced from the specification in Figure 5

The lexical rule in Figure 6 licenses the topmost V node in Figure 2 and introduces the neg predication in Figure 3. This rule is expressed in the DELPH-IN Joint Reference Formalism (called tdl; Copestake, 2002a), which can be used to implement HPSG-style typed feature structures. A grammar encoded in this way can be loaded into DELPH-IN processing tools like the LKB (Copestake, 2002b) and ACE (Crysmann and Packard, 2012) for parsing and [incr tsdb()] (Oepen, 2001) and FFTB (Packard, 2015) for treebanking.

Previous work in the AGGREGATION Project has produced grammar specifications that contain a lexicon of nouns and verbs, morphological rules and descriptions of the language's word order and case system as well as case frames for individual words. The lexicon and morphotactic rules are inferred using MOM (Wax, 2014; Zamaraeva, 2016), which we describe in Sections 4.2 and 4.3. These rules abstract away from morphophonology, so the inferred grammars are tested by parsing the morpheme-segmented line of the IGT. Inference algorithms for basic word order and case system were developed by Bender et al. (2013) and this inference together with lexical inference was used to generate grammars by Bender et al. (2014) and Zamaraeva et al. (2019a).

In this work, we present BASIL, an inference system that extends the number of phenomena that can be inferred by building on the existing morphotactic and syntactic inference systems. This system, also described in Howell 2020, infers additional lexical items including determiners, case-marking adpositions, coordinators and auxiliaries as well as properties including argument optionality, sentential negation and coordination. We also integrate syntactic and morphological inference to handle person, number and gender information on nouns, agreement between verbs and their arguments, and tense, aspect and mood contributed morphologically or by auxiliaries. Finally, whereas previous work has either evaluated the correctness of the grammar specifications on a variety of languages (Bender et al., 2013; Howell et al., 2017) or grammar performance on a single language (Bender et al., 2014; Zamaraeva et al., 2019a), we evaluate our system on grammar performance using 14 genealogically and geographically diverse languages.

4 Methodology: Inferring Grammar Specifications

This section focuses on our approach to inferring the grammar specifications illustrated in the previous section. We take as our starting point the system of Zamaraeva et al. (2019a) which integrates the morphological inference module (called MOM; Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017) and a module for inference of a few syntactic properties (Bender et al., 2014; Howell et al., 2017). To this integrated system we add extended inference for morphologically marked syntactic and semantic features, additional lexical classes and further syntactic properties to create BASIL, Building Analyses from Syntactic Inference in Local languages. BASIL takes an enriched (using INTENT; Georgi, 2016) corpus of the Xigt (Goodman et al., 2015) data type as input and produces a grammar specification file which can be input into the Grammar Matrix to generate a custom grammar for the language. This grammar specification (§4.1), often referred to as a 'choices file' in the Grammar Matrix literature, contains specifications for a lexicon (§4.2), a collection of morphological rules (§4.3), definitions of syntactico-semantic features (§4.4) and definitions of syntactic properties (§4.5) for the language at hand. During development, we used a set of 9 core languages to design and tune BASIL's algorithms and consulted an additional 18 languages that were illustrative of particular phenomena we wished to test (see §5.1). In this section, we describe each of BASIL's inference modules, including the typological range covered, what specifications the Grammar Matrix customization system requires, and how we infer appropriate specifications for a language based on IGT.[6]

4.1 The Grammar Specification

In this section, we give a brief quantitative overview of the space in which the inference system is operating. The grammar specification contains definitions for lexical items, morphological rules, syntactico-semantic features and syntactic rules. These take the form of features with either fixed or open-ended values, depending on the linguistic characteristics being defined. While a number of phenomena can be defined in the Grammar Matrix, BASIL focuses on a particular subset of lexical items and syntactic phenomena, which are modeled by 50 fixed features with 136 possible values in addition to a number of open-ended features, which allow the user to enter any value they like, rather than requiring them to choose from a menu. For some features, multiple values lead to similar coverage in the resulting grammars, so we simplify the system by focusing on a subset of the possible values. Other values are difficult to infer with sufficient accuracy from the available data or are so typologically rare that they are more likely to be inferred in error than correctly. For these reasons, BASIL targets only 99 of the 136 values, as summarized in Table 1.

While individual lexical entries and morphological rules have features that must be selected from a menu with a fixed set of values, the number of lexical items

[6] A more detailed description of these modules and the algorithms they use can be found in Howell 2020.

Phenomenon	number possible values	number targeted by inference
noun lexical entry	4	2
verb lexical entry	4	2
auxiliary lexical entry	6	4
adposition lexical entry	3	3
morphological rule	5	5
person	9	8
tense	2	1
word order	10	9
determiner order	4	4
auxiliary order	9	9
case system	9	3
argument optionality	18	15
sentential negation	41	23
coordination	12	11
total	136	99

Table 1: The number of possible values for the 50 features with a fixed value set in the grammar specification and those targeted by the inference system, broken down by syntactic category

and morphological rules defined by BASIL depends on the number of forms attested in the training corpus. Thus the size of the lexicon and morphology sections of the grammar specification varies depending on both the morphological complexity of the language and the diversity and number of samples in the training corpus. Similarly, many of the syntactico-semantic features supported by the Grammar Matrix allow the definition of unbounded numbers of possible values. For case, person, number, gender, tense, aspect and mood, we[7] compiled a list of 116 common values from the Leipzig Glossing Rules (Bickel et al., 2008), the ODIN corpus (Xia et al., 2016), Unimorph (Sylak-Glassman et al., 2015), the GOLD Ontology (GOLD, 2010) and our own observation, which the inference system can add to grammar specifications.

4.2 The Lexicon

The most accurate and fully detailed typological specification cannot produce a working grammar without a lexicon. At the same time, decent coverage over unseen texts for languages with any morphological complexity requires a lexicon built in terms of lexical entries for roots plus some model of morphological processes. The Grammar Matrix customization system elicits, as part of its input grammar specifications, descriptions of lexical classes and lexical rules. In this section, we describe lexical class specifications and how we infer them.

In brief, a lexical class is defined in terms of its part-of-speech, any further features specific to the class, and a set of lexical entries, which give the orthographic representations and semantic predicate symbols[8] for entries in that class. As an example, Figure 7 illustrates a lexical class for a type of common nouns in Meithei [mni].

```
section=lexicon
noun1_name=noun1
noun1_feat1_name=person
noun1_feat1_value=3rd
noun1_det=opt
noun1_stem1_orth=kekrú
noun1_stem1_pred=_blackberry_n_rel
noun1_stem2_orth=khoy
noun1_stem2_pred=_bee_n_rel
```

Figure 7: The definition of a common noun lexical class for Meithei

The Grammar Matrix customization system interface provides for nouns, intransitive verbs, transitive verbs, clausal complement verbs, auxiliaries, copulas, determiners, case-marking adpositions, and adjectives in its lexicon section. In addition, sections for particular syntactic phenomena allow for the definition of lexical entries for such items as conjunctions, subordinating conjunctions, complementizers, and negation adverbs. This classification of basic types of words brings with it a set of assumptions about what word classes exist in the world's languages, for example, that nouns and verbs are distinct cross-linguistically. We make no claims regarding the actual parts of speech of the lexical items MOM and BASIL infer, but attempt to model these words effectively in the resulting grammar. (For recent work showing that even languages with apparent category flexibility can be fruitfully analyzed in this way, see Crowgey's 2019 study of Lushootseed [lut].)

BASIL infers only a subset of the lexical categories supported by the Grammar Matrix, which are shown in Figure 8. In this section, we describe the process of extracting these definitions from the IGT corpus, with a focus on nouns and verbs and their subcategorization.

4.2.1 Noun and Verb Extraction

At the highest level of abstraction, lexical inference involves the definition of classes of words and the allocation of words to classes. In our system, the first pass classification of words involves parts of speech. The next level concerns inflection classes: which words

[7] This list comes from joint work with Olga Zamaraeva.

[8] We use the DELPH-IN convention for predicate symbols which includes a lemma followed by the part-of-speech (Flickinger et al., 2014a). For ease of evaluation in our current context, we use English glosses as the lemmas. For most applications, it is better to use lemmas from the language being modeled instead, as one cannot expect perfect word-level translational equivalence across languages.

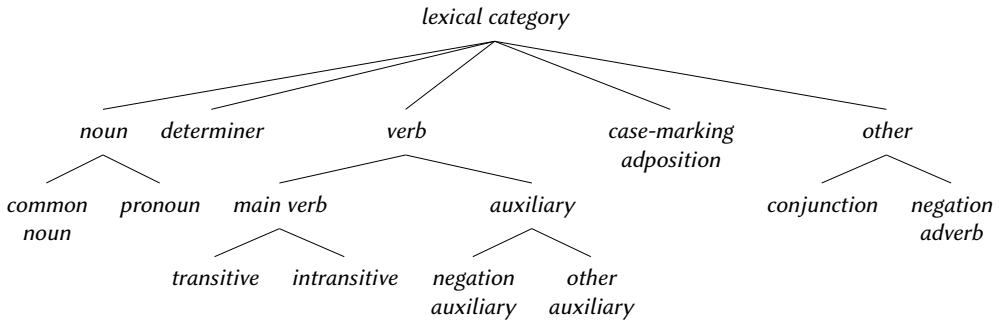


Figure 8: A taxonomy of the lexical categories that BASIL infers, organized according to the inference process

(within a part of speech) can be input to which lexical rules. To define these classes for nouns and verbs, we leverage the MOM morphological inference system. MOM identifies nouns and verbs based on their POS tags and uses a graph-based approach to identify and define inflection classes. (The morphotactic inference is further described in Section 4.3.)

4.2.2 Noun and Verb Subcategorization

In addition to defining lexical classes based on their morphotactic patterns, we must also group lexical entries based on their syntactic properties. In principle, this grouping can either be included in the input to MOM or performed on the output. Zamaraeva et al. (2019a) take the former approach to subcategorize verbs based on their valence properties by first inferring verbal case frame and including this information in MOM's input. MOM does not merge verbs with different valences, so the lexicon it produces includes separate classes for e.g. intransitive and transitive verbs, and those classes are further subcategorized based on their morphotactics.

To account for pronouns separately from common nouns and auxiliaries separately from verbs, we take the lexical classes in MOM's output and divide them based on their glosses: BASIL identifies nouns whose predication (in MOM's output) includes either an English pronoun or person, number, gender (PNG) or case features with no lemma and moves them into new lexical classes. BASIL constrains all common noun lexical classes to be third person, leaving number to the morphological analysis and inherent gender to future work (as shown in Figure 7 above). Pronoun lexical classes have more varied PNG and case values than common nouns, which BASIL accounts for by identifying any PNG and case glosses in MOM's output predication and specifying them as features on the pronoun's lexical entry.

Extracting auxiliaries from the verbal lexical classes and accounting for them in the grammar specification requires information regarding the auxiliary's syntactic distribution. For this reason, BASIL identifies auxiliaries from the source IGT rather than from MOM's lexicon, as we will describe in Section 4.5.1.

4.2.3 Additional Lexical Items

The Grammar Matrix does not support morphological inflection for determiners or adpositions, so it is not advantageous to infer these using MOM. Instead, BASIL extracts the full form orthographic representation and PNG and case features from the IGT. Where possible, we identify determiners from the POS tags, and if those are not available, BASIL looks for specific grams or lemmas in the gloss. Our grammars also support negation and coordination particles, which are described in their respective subsections of Section 4.5.

4.3 Morphotactics

The morphological component of a machine-readable grammar ultimately needs to account for which morphemes can co-occur and in which order, what the syntactic and semantic contributions of each morpheme are, and the morphophonological processes that relate the actual word forms to the collection of morphemes that make them up. The Grammar Matrix abstracts away from the morphophonology, assuming that the generated grammars will be interfaced with an external morphophonological analyzer (Bender and Good, 2005).[9] Accordingly, our inference system is only concerned with morpheme order, co-occurrence, and syntactico-semantic contributions.

The grammar specification files handle morpheme co-occurrence in terms of position classes (PCs), each of which specify what they can attach to (their 'input'),

[9] In brief, the idea is that morphophonological phenomena are best handled with different formal approaches than morpho-syntactic ones, so a parser using our grammars would be pipelined with bidirectional morphophonological analyzers. These latter map between surface realizations and morphophonolgically regularized sequences of morphemes, such as what is often found in the morpheme segmented line of IGT.

```
section=morphology
  noun-pc1_name=noun-pc1
  noun-pc1_order=suffix
  noun-pc1_inputs=noun1
     noun-pc1_lrt1_name=noun-pc1_lrt1
        noun-pc1_lrt1_feat1_name=case
        noun-pc1_lrt1_feat1_value=nom
        noun-pc1_lrt1_lri1_inflecting=yes
        noun-pc1_lrt1_lri1_orth=-pə
```

Figure 9: The definition of a position class for Lezgi

whether they are prefixes or suffixes, and which lexical rules they house. The lexical rules are defined in terms of lexical rule type (LRTs) which bear type constraints (feature/value pairs) and which in turn are instantiated by lexical rule instances (LRIs), which have specific affix spellings or are flagged as zero affixes (non-spelling-changing rules) (Goodman, 2013). An example of the specification for a position class in Lezgi [lez] is shown in Figure 9. Each PC must have at least one input (a lexical class or another PC) and a position (prefix or suffix)[10] and can be marked obligatory. Each PC must also have one or more LRTs, which can specify features on the word or on the arguments of the word. Each LRT must have one or more LRIs, which includes an orthographic form or a flag indicating that the rule involves no overt morpheme.

We use the MOM morphotactic inference system (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017, 2019a) to infer the morphological rules. MOM infers a graph of the morphemes by collecting the affixes for each word with a noun or verb POS tag, creating a PC with an LRT which includes any features found in the gloss and an LRI with the appropriate orthographic representation and merging PCs that have overlapping inputs.[11]

While the morphotactic graph is essential for processing individual words, the morpho-syntactic or morpho-semantic features on those morphemes are key to producing the correct parse for larger phrases and sentences. MOM uses a feature dictionary comprising a large number of known glosses, grouped by their type, to map common grams to features. For example, the grams 'IPFV', 'IMPFV' and 'IMPERF' are all mapped to imperfective aspect. When MOM constructs the lexical rule types, it adds the features corresponding to any PNG, TAM or case grams to the lexical rule.

Non-inflecting lexical rules pose a particular challenge because they are not typically glossed as separate morphemes in IGT but rather indicated with a gram attached to the previous element with a ".", if they are indicated at all. MOM only creates non-inflecting rules for glosses it is able to map to PNG, case or TAM features, and only when such a gloss is found attached to the gloss for a stem. For example, if a noun is glossed as 'dog.NOM', MOM creates a non-inflecting lexical rule to add nominative case. All PCs which contain a non-inflecting LRI are made obligatory, so that forms without overt affixes do not end up only optionally bearing the features associated with that part of the paradigm.[12]

The result of morphological inference with MOM is a set of lexical rules grouped into position classes modeling their combinatorial potential. Within those position classes are lexical rule types that contribute features and in turn contain lexical rule instances, which either correspond to a particular orthography or are non-inflecting. Both the morphological rules in this section and the lexical entries in Section 4.2 contain morpho-syntactic features which interact with the syntactic inference in Section 4.5. The next section is concerned with how we define those features in the grammar specification, so that they will interact properly in the resulting grammars.

4.4 Syntactico-semantic Features

A great deal of semantic information is expressed morphologically in the form of person, number and gender (PNG) marking on nouns or agreement on verbs and tense, aspect and mood (TAM) inflection on verbs and auxiliaries. In order to model these features, the grammar specification must contain two types of definitions: First, the features and values themselves must be defined as belonging to the appropriate PNG or TAM category; and second, they must be associated with the appropriate lexical entries or morphological rules. The work of associating these features with the appropriate forms was described in Sections 4.2 and 4.3. When building the lexicon and morphological rules, MOM associates each feature value (e.g. perfective) with a type (e.g. aspect) according to their classifications in the GOLD Ontology (GOLD, 2010) and Unimorph (Sylak-Glassman et al., 2015). In this section we describe how BASIL uses these features and types to define more detailed type definitions for each PNG and TAM category, so the syntactic constraints contributed by these features can be used in the grammar and their semantic contributions will be reflected in the semantic representations.

[10] The Grammar Matrix does not handle circumfixes separately. These must be specified as individual prefixes and suffixes. Infixes are not explicitly handled; instead the Matrix assumes that a morphophonological analyzer regularizes these to prefixes or suffixes. See footnote 9.

[11] For more detail, see op cit.

[12] The addition of non-inflecting lexical rules to MOM, as well as the functionality of collecting the initial set of grams and adding features to lexical rules described in the preceding paragraph, is from unpublished work by Olga Zamaraeva.

4.4.1 Person

Generally speaking, person is a feature that marks the entities in an utterance with respect to discourse participants (Siewierska, 2004), where *first* is the speaker, *second* is the addressee and *third* is someone or something outside of the discourse context. Combinations of these persons, such as *first+second* 'I and you' and *first+third* 'I and they' are sometimes given special grammatical treatment and are often referred to as *inclusive* and *exclusive* (Cysouw, 2013). The Grammar Matrix's library for person (Drellishak, 2009) provides a set of six options for person distinctions: first, second, third; first, second, third and fourth; first and non-first; second and non-second; third and non-third; and none. It also allows three options with regard to subtypes in the first person: none, inclusive vs. exclusive (along with the number categories in which this distinction applies) and other.

After collecting all of the person features from the lexical items and morphological rules, BASIL posits that the language contains first, second, third and fourth person if it found 4th person; first, second and third person if it found 3rd and either 1st or 2nd; and then first and non-first if it found 1st; second and non-second if it found 2nd; third and non-third if it found third; and otherwise none. BASIL then checks for inclusive and exclusive features and if it finds any, it defines an inclusive/exclusive distinction.

4.4.2 Number

Number indicates how many entities are being referred to. If a language marks number at all, this distinction can be as simple as singular vs. plural or may be more modular distinguishing dual (two), paucal (a few) and other numbers of entities (Corbett, 2000). The numbers distinguished by a language vary cross-linguistically and it is possible for these features to form a hierarchy (e.g. non-singular might subsume dual and plural). Thus, the Grammar Matrix allows number features to be freely added to the specification file, forming a hierarchy if desired (Drellishak, 2009). BASIL defines a number value for each of the numbers found in the morphology and lexicon. Currently, it defines each of these as sister types, rather than inferring a hierarchy of supertypes and subtypes, which we leave to future work.

4.4.3 Gender

Gender is another fairly open-ended category in the world's languages. While some languages like Russian [rus] distinguish just masculine, feminine and neuter, Bantu languages such as Kiswahili [swh] distinguish a complex system of genders (Corbett, 1991). Linguists also vary in their annotation of gender features either using grams like M or MASC or using numerals for more complex systems. To accommodate this flexibility in the gender distinctions in language and linguists' annotation preferences, the Grammar Matrix allows the addition of any number of genders by any name, and allows the specification of a hierarchy (e.g. to support agreement markers that are ambiguous between two or more gender values). As with number, BASIL defines a gender value for each of the genders found in the morphology and lexicon, but does not infer a hierarchy.

4.4.4 Tense, Aspect and Mood

Every language has some grammatical expression of time, which falls into the categories of tense, aspect and/or mood, and these features can be marked either morphologically on the verb, with an auxiliary or morphologically on an auxiliary, and a single utterance may include a combination of these expressions (Hopper, 1982).[13] For example, in the IGT from Matsigenka [mcb] in (3), the verb *oataira* is marked with regressive aspect (REG) and realis mood (REALIS), while the verb *oponiakara* is marked with perfective (PERF) aspect and realis mood (REALIS). Michael (2008) characterizes the regressive aspect as a subtype of perfective aspect that indicates motion back to a salient point of origin.

(3) ovashi oataira
ovashi o-a-t-a-i=ra
so 3fS-go-EPC-REG-REALIS=SUB
oponiakara.
o-poni-ak-a=ra
3fS-come.from-PERF-REALIS.REFL=SUB
'Then she went back to where she came from.' [mcb] (Michael et al., 2013)

The TAM categories contain a number of possible values cross-linguistically and, as illustrated by the regressive and perfective aspects described by Michael, can form hierarchies. As with the number and gender libraries, the TAM library of the Grammar Matrix (Poulson, 2011) also allows the definition of any number of values for each of tense, aspect and mood and also allows the definition of hierarchies. BASIL defines each TAM feature as either tense, aspect or mood in the respective section of the grammar specification, leaving the inference of hierarchies to future work.

4.4.5 Summary

We described six categories of syntactico-semantic features: person, number, gender, tense, aspect and mood. These features are added to the specifications of lexical

[13] In NLP, the TimeML specification language (Pustejovsky et al., 2003) has been used in an effort to standardize such expressions of time, and has been made more cross-linguistically viable by efforts such as Zymla 2017.

entries or morphological rules according to the methodologies described in Sections 4.2 and 4.3 and defined as belonging to their respective categories. The result of these definitions is a grammar that produces semantic representations that contain this information and enforces agreement between heads and their arguments.

4.5 Syntactic Properties

In this section, we provide a high-level description of the algorithms used for inferring each of the syntactic phenomena accounted for in our grammars. Using the projected dependency tags provided by INTENT and typologically-informed heuristics, we make generalizations about distributional properties of the language and posit the appropriate definitions for that grammar specification for a range of syntactic phenomena. These include broad-brush, language-level properties (e.g. 'the case alignment is ergative-absolutive'), properties associated with specific constructions (e.g. 'this form can coordinate VPs in a monosyndetic pattern') and specific lexical items (e.g. 'negation is marked via an auxiliary with this orthography that combines with a VP and raises the subject').

4.5.1 Word Order and Auxiliaries

Languages vary in both their degree of word-order flexibility and, if only specific orders are allowed, which ones are (e.g. Dryer, 2013c). When linguists talk about the 'word order' of a language, they are frequently referring to the relative order of a verb and its arguments (subject, complement), but there are also cross-linguistic differences in the order of determiners (if present) with respect to their head nouns, adpositions with respect to NPs, and others. The 'word order' section of a Grammar Matrix grammar specification takes information about each of these (Bender et al., 2010).

We adopt the approach of Bender et al. (2013), which maps constituent word orders observed in the data to one of ten canonical word orders (SOV, SVO, OSV, OVS, VSO, VOS, v-initial, v-final, v2 and free). This approach identifies verbs based on their POS tags and their subjects and objects using projected dependency labels. Each observed order of verbs and subjects, verbs and objects and subjects and objects is counted to compute a three dimensional vector representing the respective order of verbs, subjects and objects in the language, which can be compared to the vector representations for each canonical word order. Following Bender et al., BASIL posits the canonical word order whose vector has the shortest euclidean distance from the observed language vector as the canonical word order for the language.

Also following Bender et al. (2013), we take a simpler approach to predict determiner-noun order. Collecting each noun and determiner pair from the projected dependencies, we count the number of observed determiners before vs. after the noun and posit whichever order is most common.

Whereas previous work did not account for auxiliaries, BASIL both identifies auxiliaries as lexical items and infers their syntactic properties. This includes identifying their position with respect to the main verb and inferring what type of constituent they attach to (a verb (V), verb phrase (VP) or sentence (S)), whether they attach before or after that constituent, and whether multiple auxiliaries are possible. We identify auxiliaries in the corpus as words that are either glossed with an English auxiliary or modal or glossed with only morpho-syntactic or morpho-semantic features and no lemma. While collecting auxiliaries from the corpus we identify the main verb and its subject and object from the projected dependencies. We use these to discover whether the auxiliary occurs before or after the main verb and check for a subject intervening between an auxiliary and verb, which would indicate that the auxiliary takes an S complement instead of a VP, or an auxiliary intervening between a verb and its object, which would indicate that the auxiliary attaches to a V, rather than a VP. If no evidence for V or S attachment is found, BASIL defaults to VP attachment, as the argument-composition analysis that the Grammar Matrix uses to model auxiliaries with V complements is computationally very expensive (see Bender 2010) and we hypothesize that S attaching auxiliaries are typologically rare.

Because the MOM morphotactic inference system infers auxiliaries as verbs when constructing the lexicon, BASIL must reclassify these lexical items to give them the proper definitions to function as auxiliaries in the grammar. BASIL does this by finding any verbs in the MOM-generated lexicon that have the same lemma as those it identified as auxiliaries. For each, BASIL defines an auxiliary lexical class that is input to the same morphological position classes and contains the same features as the verb lexical class inferred by MOM. Because auxiliaries are often homophonous with main verbs, BASIL does not remove the main verb lexical entry.

In addition to the lemma, feature and morphological combinatorial information described above, the Grammar Matrix requires specifications for the semantic contribution of the auxiliary. When BASIL constructs the auxiliary lexical items from verb lexical items inferred by MOM, it specifies the auxiliary as semantically contentful and adds the predication value from the verb if the original verb's predication contains an English lemma (e.g. _should_v_rel), rather than containing only grams for syntactico-semantic features. BASIL also adds a negation predication if the auxiliary contributes negation (see Section 4.5.4 for negation inference).

Finally, the lexical entry includes a value for the case of its subject, which can be specified as a specific case, no case restrictions, or the case assigned by the verbal complement. With our development languages, we tested an algorithm in which BASIL checks for differences in the case on subjects in sentences with and without auxiliaries, and adds this constraint to the lexicon. We found that this inference is frequently confounded by other factors that can affect the subject's case, so we did not include this inference in BASIL and leave a more accurate algorithm to future work. Currently BASIL posits no case restrictions if A) the language does not have a case system or B) the auxiliary always occurs with a different case than the one inferred for the verb's case frame (this leads to some ambiguity, but avoids the loss in coverage that results from positing a case that was assigned due to other syntactic factors). Otherwise it posits that the auxiliary takes its case restrictions from the main verb.

After identifying the auxiliaries in the corpus, we allow for a post-hoc change to the main word order to account for second position clitic clusters. The Grammar Matrix supports an analysis set forth by Bender (2008c) of second position clitics/clitic clusters as auxiliaries in a V2 language, when those clitics express TAM and/or agreement features. Clitic clusters that contain PNG agreement and TAM information are identified during auxiliary inference and if they occur overwhelmingly as the second word of each sentence, BASIL posits V2 word order for the language to leverage this analysis.

4.5.2 Case System and Case Frame

A language which marks case has variations in the forms of the noun phrases correlated with their function in the sentence (Comrie, 1989; Dixon, 1994). A typical case system will involve both the case required of core arguments of typical verbs, as well as additional cases used when NPs function as modifiers (e.g. locative case) and sometimes selected for idiosyncratically by specific verbs. Case systems are differentiated according to the alignment they provide for the core arguments of intransitive and transitive verbs. The Grammar Matrix customization system's case library (Drellishak, 2009) provides nine overarching case systems (core argument case alignments) and facilitates defining any number of additional cases. The selection of the core case system enables default case frames for each verb type, but grammar specifications can also bypass these and define verb types which leave case underspecified or select for alternate case patterns.

To infer the overarching case system, we use an algorithm developed by Bender et al. (2013) and reimplemented to use an enriched Xigt corpus by Howell et al. (2017), which uses a simple heuristic based on the total counts of known case grams in the data. This approach only infers four case systems: nominative-accusative, ergative-absolutive, split-ergative and none. Because split-ergative requires information about the nature of the split, we map it to ergative-absolutive. In addition to inferring the overarching case system, we also collect any other case grams in the corpus and define these in the grammar specification, so that we can also handle verbs that require alternate case frames. Here we infer only intransitive and transitive verbs, leaving ditransitive (which are not currently supported by the Grammar Matrix) and clausal complement-taking verbs to future work.

To find the case frame of each intransitive and transitive verb in the corpus, BASIL uses the dependency parse of the English sentence to identify verbs that have zero or one direct object, skipping any that are passive or have an indirect object or clausal complement (following Zamaraeva et al. (2019a), such verbs will be excluded from the final grammar). We find the case of the subject and object in the gloss line and if no case gram is found in the gloss, we posit default case based on the overarching case system. In cases where the marked case doesn't match the default, we posit the attested case for that verb's arguments. Our approach is similar to that of Zamaraeva et al. (2019a), but differs in that we use projected dependency parses rather than phrase structure trees and that we account for verbal case frames that differ from the overarching system.

These constraints interact with the case features on noun-phrases when verbs unify with their arguments. Case features may be licensed by the morphological rules on nouns which were inferred by the morphological component described in Section 4.3, can be lexically specified (e.g. for pronouns, see Section 4.2.2) or can be indicated by the determiner or a case-marking adposition. If, for example, the feature specification [CASE acc] is associated with a lexical rule attaching an accusative case marker to a noun, or if [CASE acc] is in the lexical entry for a determiner or adposition, NPs or PPs built with these lexical entries or rules will be incompatible with argument positions that require [CASE nom].

Having described the inference algorithms and systems for phenomena such as morphotactics, word order and case, and the ways in which we refined, adapted and added to them, we now turn to the entirely new inference modules that we contribute in this paper, beginning with argument optionality.

4.5.3 Argument Optionality and Marking of Arguments on Verbs

Languages vary in the extent to which and under what conditions they allow dropped arguments: some languages allow core arguments of any verb to be dropped freely, while others are more restrictive if argument

dropping is possible at all. These restrictions range from the specific verbs for which argument dropping is allowed, subject vs. non-subject arguments, specific syntactic contexts (e.g. only in certain tenses), or whether the verb is required to agree with overt vs. dropped arguments (Ackema et al., 2006; Dryer, 2013a). The Matsigenka example in (4) shows a verb with no overt arguments that is inflected for agreement with both the subject and object.[14]

(4) oogaigavakari
o-og-a-ig-av-ak-a=ri
3FS-eat-EPV-PL-TRNS-PERF-REALIS.REFL=3MO
'She ate them.' [mcb] (adapted from Michael et al., 2013)

The Grammar Matrix accounts for subject and object dropping as either lexically licensed (allowed for certain verbs) or possible for any verb (Saleem, 2010; Saleem and Bender, 2010). It also allows argument dropping to be constrained by agreement markers on the verb which can be optional, required or not allowed when the subject/object is overt, and similarly when the subject/object is dropped. Finally, specific syntactic contexts in which subject dropping is possible can be defined. Our inference focuses on determining whether argument dropping is permitted for subjects and objects in a language and leaves constraints on the context to future work. We infer whether agreement is required for dropped vs. overt arguments, which requires differentiating subject agreement markers and object agreement markers; however, we leave the integration of this inference with the morphological rules that license agreement to future work.

In order to identify whether subject and/or object dropping is possible in the language, BASIL begins by collecting all of the transitive and intransitive verbs[15] in the corpus together with their overt arguments, based on the projected dependencies as it did for case-frame inference (§4.5.2). Whereas the case-frame inference methodology determines if a verb is transitive based solely on the presence of an overt object in the English translation, here we account for the fact that some English verbs allow object dropping. If the corresponding verb in the English translation has a direct object, we assume that the verb is transitive. If no object is found, BASIL cross-references the verb's gloss with a list of English object-dropping verbs from the lexical entries in the English Resource Grammar (ERG v. 1214; Flickinger, 2000, 2011) of the type v_np*. If the verb is found in this list, BASIL posits that the verb is transitive and otherwise intransitive. Although the argument optionality of verbs does not necessarily map across languages, leveraging this list of English object-dropping verbs allows us to err on the side of positing transitivity, and we find that doing so improves the coverage of the resulting grammars.

Agreement with the subject or object can be marked either on the main verb or on an auxiliary. To determine whether a verbal complex has subject and/or object marking, BASIL identifies any auxiliaries associated with each verb and collects all agreement markers (across the verb and any auxiliaries), using a hand-compiled list of common agreement glosses. We compiled this list from the agreement glosses used by MapGloss (Lockwood, 2016) as well as observed glosses in the development data. Although agreement is not the only way arguments are marked on verbs (for example, in Hausa the verb's inflected form depends on whether or not an overt object is present, but this form does not include any PNG information (Newman, 2000)), it is the most common form and the easiest to identify. In addition to collecting all agreement markers, we use a heuristic to identify whether the agreement markers correspond to more than one argument: if the set of agreement glosses has multiple glosses of a particular category (e.g. person, number or gender), BASIL says that the verb is marked for more than one argument. This approach is particularly valuable when a single morpheme is used to mark two arguments. For example in (5) from Basque [eus], *dio* is glossed as 3ABS-3DAT.3ERG, containing three third person glosses, so BASIL counts three agreement glosses on that verb.

(5) Eduk neska Toniri aipatu
Edu-k neska Toni-ri aipatu
Edu-ERG girl.ABS Toni-DAT mention
dio
d-io
3ABS-3DAT.3ERG
'Edu has mentioned the girl to Toni.' [eus] (adapted from Xia et al., 2016)

We use the presence of agreement features on any verb in the set to detect argument marking on the main verb. Intransitive verbs with any agreement gloss are classified as having subject marking. The orthographies associated with these glosses are saved in a set of known subject markers. After all of the subject markers on intransitive verbs have been collected, BASIL looks at the transitive verbs. Transitive verbs with more than one agreement gloss (like that in (5)) are classified as having subject and object marking. Transitive verbs with only one agreement gloss which corresponds to the orthography of a known subject marker are classified as having subject marking and the remainder are classified as having object agreement. The set of known

[14] We analyze the pronominal clitics in Matsigenka as affixes, rather than independent words, following Inman (2015).

[15] Because BASIL does not infer ditransitive or clausal complement-taking verbs, it excludes them from consideration when inferring argument dropping.

subject glosses is included in the input to MOM. When deciding if a PNG gram should be identified with the subject or object, MOM consults this list and associates it with the subject if the verb is intransitive or the morpheme is in the set of subject morphemes and with the object otherwise.

BASIL's inference for argument optionality has two components: (1) inferring whether subjects and objects can be dropped, and (2) inferring whether argument marking on the verb is possible or even required when arguments are dropped or overt. The latter involves identifying argument markers in the form of agreement morphemes and discriminating between subject and object agreement markers. Our approach focuses on increasing the coverage of the inferred grammars, while future work to enforce or prohibit argument marking on verbs with overt versus dropped arguments would decrease ambiguity.

4.5.4 Sentential Negation

All human languages have a means of expressing sentential negation, but they vary in how many markers are used and whether those markers are independent words, bound morphemes (Östen Dahl, 1979; Dryer, 2005, 2013b; Miestamo, 2008) or a missing morpheme in the paradigm, such as the absence of a tense marker indicating negation in some south Dravidian languages (Master, 1946). Crowgey (2012) models sentential negation in the Grammar Matrix, allowing it to be marked with 0, 1 or 2 morphemes (calling these strategies *zero, simple* and *bipartite*), which can be bound morphemes, syntactic heads (auxiliaries) or uninflected particles (adverbs). The analyses provided by the Grammar Matrix ensure that there is only one negation predication in the semantics, regardless of the number and type of markers in the strategy. BASIL infers each of the possible combinations as described below.

We first identify sentences with sentential negation based on the English translation and then target the gloss line of the IGT to find negation morphemes, based on common glosses, such as 'NEG' and 'not'. BASIL considers glosses on affixes to be inflectional negation. We expect that zero-marked negation will be annotated with a negation gloss on a stem or on another morpheme and will therefore be modeled with a non-inflecting lexical rule as described in 4.3, so BASIL accounts for it using the morphological negation specification. If inflectional negation is detected, this is indicated in the sentential negation portion of the grammar specification which in turn enables a negation pseudo-feature which can be added to lexical rules. The distributional properties for negation affixes (including zero-negation) are inferred and specified by the morphological inference system in Section 4.3, which puts the negation pseudo-feature on the appropriate lexical rule. The Grammar Matrix customization system interprets this pseudo-feature and ensures that the resulting lexical rules carry negation semantics, as shown in Figure 6.

A root glossed as negation could be either an auxiliary or an adverb. The English dependency parse does not help us decide which, as it simply encodes facts about negation in English. Instead, we compare these negation words with the auxiliaries collected in Section 4.5.1. If auxiliary entries were inferred for orthographies glossed for negation, we treat them as such. Otherwise we define them as adverbs. The distributional properties of negation auxiliaries were inferred as part of auxiliary inference (§4.5.1), so there is no additional work to be done. In the case of negation adverbs, we use the same process as we did for auxiliaries to decide what type of constituent they attach to (VP or S) and whether they occur before of after that constituent.

After identifying instances of sentential negation in the corpus, BASIL compares the number of sentences that include one negation marker with those that include more than one negation marker. Although BASIL only looks at sentences with sentential negation, it does not distinguish between sentential and constituent negation markers, and can mistake a negated sentence with additional constituent negation as bipartite negation. However, we seek to avoid confounding from constituent negation co-occurring with sentential negation by taking the most common strategy (simple or bipartite) found in the corpus.

If simple negation is the most common, the Grammar Matrix lets us add all of the strategies we found (affix, auxiliary, and adverb) to the grammar specification. For bipartite negation, we can only specify one combination of markers, so if bipartite negation was the most common strategy found in the corpus, we add the two most common co-occuring types of negation markers (e.g. adverb and affix) to the grammar specification. While the Matrix only allows us to add one orthography for a negation adverb (so we use the most common), we are able to specify as many negation affixes and auxiliaries as we find in the corpus.

4.5.5 Coordination

Coordination is possible for a wide range of constituent types, called coordinands, and can be marked with either free or bound morphemes, called coordinators. Coordinators can attach to all (omnisyndetic), all but one (polysyndetic), one (monosyndetic) or none (asyndetic) of the coordinands (Drellishak, 2004; Haspelmath, 2007). The Grammar Matrix models all of these possibilities and allows us to define any number of strategies for nouns, noun phrases, verbs, verb phrases and sentences (Drellishak and Bender, 2005).

As with sentential negation, BASIL identifies IGT that exhibit coordination based on the English transla-

tion and then finds the coordinators first by looking for the word aligned by INTENT with the English coordinator and then, because alignment isn't always successful, by looking for the glosses 'COORD', 'CONJ', 'CCONJ' and 'and'. Then BASIL uses the projected dependencies to collect the dependent of each coordinator and these dependents are assumed to be the coordinands. As a fallback, if BASIL cannot find coordinands via projected dependencies, it looks for them by collecting the words that occur in between coordinators, although this approach is less successful for monosyndetic coordination. BASIL then compares the number of coordinators and coordinands to decide if the sentence exemplifies asyndetic, monosyndetic or omnisyndetic coordination. Differentiating between mono- and polysyndetic coordination is rather difficult as most examples in the corpora only have two coordinands, and the construction 'A and B' could be either mono- or polysyndetic. However, monosyndetic coordination can be used to model polysyndetic (e.g. [[A and B] and C]), so BASIL defaults to monosyndetic in cases that might be mono- or polysyndetic.

For each coordination strategy, we also identify the lexical category of the coordinand (noun or verb) and use heuristics to decide at what level the coordination takes place (word or phrase in the case of nouns and word, phrase or sentence for verbs). Because the Grammar Matrix allows any number of coordination strategies, we add each distinct coordination strategy that we detect in the corpus to the grammar specification.

4.6 Summary

In this section we described four types of inference that produce the necessary components of our inferred grammar specifications: lexical, morphotactic, morpho-syntactic/morpho-semantic and syntactic. For inference of noun and verb lexical classes and lexical entries, we rely primarily on the MOM morphotactic inference system, but make new contributions to lexical inference in the form of auxiliary, adposition and determiner inference as well as lexical types defined as part of syntactic inference such as negation adverbs or coordinators. We also leverage MOM to infer morphological rules for nouns and verbs, and build on the system by improving the detection of subject and object agreement, as described in Section 4.5.3, and adding the definitions of PNG and TAM features to the grammar specification, so that these syntactico-semantic features can be included in the semantic representations. We built on previous algorithms for inferring syntactic properties such as word order and case and added new algorithms for argument optionality, negation and coordination.

The scope of this inference spans a large number of feature-value pairs in the grammar specification, as we illustrate in Table 1, and testing the inference for all of these on real data would require a vast set of datasets from typologically diverse languages. At the same time, it is possible that specifications allowed by the Grammar Matrix or targeted by BASIL are not sufficient to correctly model some languages. In the following section, we describe our data-driven approach to development in which we considered corpora from a wide range of diverse languages and from a variety of data formats to develop and test the algorithms detailed in this section.

5 Development Languages

We developed the inference algorithms described in Section 4 using a data-driven approach in which we consulted the typological literature for each phenomenon and actively tested each algorithm on a diverse set of languages throughout implementation. In this section, we describe the languages and datasets we used during development (§5.1), phenomena that appear in our datasets, both targeted by BASIL and otherwise (§5.2) and BASIL's performance on the development datasets (§5.3).

5.1 Dev Languages and Datasets

In order to thoroughly test BASIL on the phenomena described in Section 4, it is necessary to use languages that are typologically varied, representing as many language families and geographic areas as possible. For development, we made use of 9 datasets for languages from 7 language families and 4 continents. In addition to these core development datasets, we tested individual phenomena using datasets from another 18 languages to span a total of 19 language families and 6 continents. These languages, their language families and details of the corpora are listed in Table 2. Their geographic distribution is shown in Figure 10, with development languages in red (1-9) and additional consulted languages in blue (10-27).[16] Held-out languages which we discuss in Section 6.3 are in green (28-32).

We selected the core development languages based on the size and quality of the dataset as well as for some of the syntactic phenomena exhibited by those languages. The majority of these corpora come from a FLEx or Toolbox corpus that was curated by a documentary linguist (or a group of linguists). To support the development and implementation of inference for specific syntactic and morpho-syntactic phenomena, we also consulted additional datasets for languages which represent those phenomena. These datasets not only contribute to the diversity of the languages we worked

[16] In most cases, these coordinates come from WALS (Dryer and Haspelmath, 2013). If information from WALS was not available, we consulted other sources, starting with descriptions of where the languages are spoken from the reference grammars we worked with.

Figure 10: Map of the coordinates where languages used in the development are spoken

	Language	ISO 639-3	Family	Source Type	Number of IGT	POS tags in source
	Development					
1	Abui	abz	Trans-New Guinea	Toolbox	1568	yes
2	Chintang	ctn	Sino-Tibetan	Toolbox	9785	yes
3	Matsigenka	mcb	Arawakan	FLEx	349	yes
4	Nuuchahnulth	nuk	Wakashan	FLEx	641	no
5	Wambaya	wmb	Mirndi	Book	818	no
6	Haiki	yaq	Uto-Aztecan	FLEx	2235	yes
7	Lezgi	lez	Nakh-Daghestanian	FLEx	1168	yes
8	Meithei	mni	Sino-Tibetan	FLEx	955	yes
9	Tsova-Tush	bbl	Nakh-Daghestanian	FLEx	1601	yes
	Consulted					
10	Bardi	bcj	Nyulnyulan	Book	178	no
11	Ik	ikx	Eastern Sudanic	Book	201	no
12	Old Javanese	jav	Austronesian	Toolbox	308	no
13	Yup'ik	esu	Eskimo-Aleut	Book	217	no
14	Basque	eus	Basque	ODIN	1033	no
15	Dutch	nld	Indo-European	ODIN	3543	no
16	Finnish	fin	Uralic	ODIN	3123	no
17	Greek	ell	Indo-European	ODIN	2065	no
18	Hausa	hau	Afro-Asiatic	ODIN	2504	no
19	Hungarian	hun	Uralic	ODIN	2077	no
20	Indonesian	ind	Austronesian	ODIN	1699	no
21	Italian	ita	Indo-European	ODIN	3513	no
22	Japanese	jpn	Japonic	ODIN	6655	no
				Book	116	no
23	Korean	kor	Korean	ODIN	5383	no
24	Mandarin	cmn	Sino-Tibetan	ODIN	5045	no
25	Polish	pol	Indo-European	ODIN	2691	no
26	Russian	rus	Indo-European	ODIN	4161	no
27	Turkish	tur	Altaic	ODIN	2617	no

Table 2: Languages used in development

with, but also to the variety of source formats and dataset styles. A number of the datasets we consulted for individual phenomena (languages 14-27) come from the ODIN corpus (Xia et al., 2016), which is a collection of IGT scraped from academic papers. We also extracted four corpora from descriptive grammars, using the pipeline for extracting IGT from text and converting it to the Xigt data model developed by Xia et al. (2016). A full list of citations for the corpora and any descriptive resources we consulted are in Appendix C.

Later in this section, we describe BASIL's coverage over the development datasets. To contextualize that discussion, we begin with an overview of the languages and their respective datasets.

Abui [abz] is an Alor-Pantar language in the Trans-New Guinea language family. It has about 16,000 speakers and is primarily spoken on the Alor island of Indonesia (Kratochvíl, 2007). This dataset (Kratochvíl, 2019) comes from a Toolbox corpus which contains about 18,000 sentences from both elicitation and transcribed speech. As part of an ongoing documentation effort, the dataset is only partially glossed. We filtered the data based on the presence of full segmentation and glossing, and removed duplicates and examples marked as ungrammatical, to create a dataset of 1,500 sentences.

Chintang [ctn] is a Kiranti language of the Sino-Tibetan family spoken in Nepal with 4,000-5,000 speakers (Schikowski, 2013). The Toolbox dataset is quite large, coming from a long-term documentation effort (Bickel et al., 2013b). We use a fully segmented and glossed subset of the data containing almost 10,000 sentences. The type of language represented in the corpus is diverse, containing transcribed conversations, ritual language, narratives and a few other genres.

Haiki [yaq] is a Taracahitic language of the Uto-Aztecan family and is spoken by about 21,000 people in Mexico and the United States (Eberhard et al., 2019). There are multiple spellings of the name of this language, including Yaqui, which is the official name of the tribe in the United States and Mexico; however, Haiki is the correct spelling in the Pascua Yaqui orthography (Sanchez et al., 2015). The corpus (Harley, 2019) is quite large with almost 11,000 IGT, but as with most ongoing projects, is only partially annotated with interlinear glosses and part-of-speech tags. After filtering IGT with no glosses and removing ungrammatical examples and duplicates, we worked with a set of just over 2,000 IGT.

Lezgi [lez] belongs to the Lezgian subgroup of the Nakh-Daghestanian language family (Donet, 2014a). It is spoken by about 400,000 people (Eberhard et al., 2019), primarily in Daghestan and Azerbaijan (Donet, 2014a). The glossing and POS tagging in this corpus (Donet, 2014b) are fairly complete, resulting in a set of over 1,100 IGT after minor filtering and removing ungrammatical examples and duplicates.

Matsigenka [mcb] is a Maipurean language of the Arawakan family spoken in Peru by about 10,000 people (O'Hagan, 2018). The FLEx corpus (Michael et al., 2013) is made up of narratives that are fully segmented and glossed. Of the approximately 5,000 IGT in the corpus, some have English translations, while the vast majority of the translations are in Spanish. BASIL relies on computational resources for English, both through its dependency on the INTENT (Georgi, 2016) system (which parses the English translation of an IGT and projects the dependency parses onto the language) and through the list of English verbs referenced in Section 4.5.3, and thus BASIL requires IGT with English translations. From the full Matsigenka corpus, we[17] identified about 350 IGT with English translations.

Meithei [mni] is a Kuki-Chin-Naga language of the Sino-Tibetan language family. It is spoken predominantly in Manipur State, but has about 56 million speakers living across a wide region, including in China, India, Nepal and Myanmar (Chelliah, 2011). The FLEx corpus (Chelliah, 2019) contains about 1,800 IGT, but as part of an ongoing documentation effort, is only partially annotated. After filtering for fully-glossed IGT and removing duplicates and ungrammatical examples, the corpus has about 1,000 items. Compared to other corpora in our development set, this corpus contains a high proportion of complex sentences, which include subordinate clauses that are not covered by inference. Nevertheless, it is a strong example for how much typological information can be learned from a corpus, even when many of the sentences contain phenomena that are beyond the scope of the inference system.

Nuuchahnulth [nuk] is Southern Wakashan language of Vancouver Island in Canada and has only about 130 fluent speakers (Eberhard et al., 2019). The FLEx dataset (Inman, 2019b) was curated in connection with a dissertation on multi-predicate constructions and contains both transcribed narratives and elicitations, many of which target this construction. The dataset includes about 650 examples which are fully glossed and segmented. Inman's corpus does not include POS tags, which are required by MOM to build the lexicon of nouns and verbs. For many IGT, these are available from the projected part of speech tags from INTENT. However, because INTENT does not always successfully find an alignment (this can be particularly challenging for polysynthetic languages), we use an additional heuristic to identify verbs. Because single-word sentences are very common in this poly-synthetic language, we supplemented the projected POS tags by pre-

[17] Most of these were identified by previous research assistants on the AGGREGATION project and more were extracted by Angelina McMillan-Major.

processing the corpus to assign a verbal POS tag to the only word in any one-word IGT if the dependency parse for the translation was headed by a verb.

Wambaya [wmb] is a West Barkly language in the Mirndi family, which has about 60 speakers (Eberhard et al., 2019). The Wambaya dataset is distinct from our other development datasets as it was extracted from the examples in a descriptive grammar (Nordlinger, 1998). As such, it does not contain linguist-provided POS tags and the possibility of alignment errors in the interlinearization is higher, due to the process of extracting IGT from text. Nevertheless, this language illustrates a number of phenomena that guided our development and the use of a descriptive grammar allows us to explore the possibility of inferring grammars to accompany descriptive resources along the lines of Bouma et al. 2015.

Tsova-Tush [bbl], also referred to by the endonym Bats or Batsbi, is a Northeast Caucasian language of the Nakh subgroup of the Nakh-Daghestanian language family (Hauk and Harris, forthcoming). It is spoken in Georgia by about 2,500-3,200 people (*ibid.*). The corpus (Hauk, 2016–2019) contains elicitation and transcribed text and the glossing and part of speech tags are almost complete, including over 1,600 IGT after removing ungrammatical examples and duplicates.

5.2 Dev Language Phenomena

In this section we quantify the degree to which the inference system was tested by the development languages described above. In Section 4.1, we described the space of the inference task in terms of the number of features and values that BASIL is designed to add to the grammar specification to account for the phenomena it handles. We identified 50 features with a fixed set of values (listed in Table 1) totaling 136 possible values in the Grammar Matrix grammar specifications that are relevant to the phenomena targeted by BASIL. Our system is designed to infer 99 of those 136 values. When inferring grammar specifications for the 9 development languages, 37 of the 50 features and 71 of the 99 values were inferred by BASIL from the development data, as detailed in Table 3. We also reported in Section 4.1 that BASIL can identify 116 morpho-syntactic and morpho-semantic features from their glosses in the IGT. 66 of those 116 features are found in the development datasets (see Table 4).

While the development languages test a significant portion of the phenomena targeted by BASIL, they do not exhaustively test every facet. For this reason, we consulted an additional 18 languages (represented in blue in Figure 10) to test as many of the feature-value pairs as possible, in order to create a system that would generalize beyond the development languages.

The phenomena targeted by BASIL (§4) are only a subset of the phenomena necessary to fully model a language or to parse all of the sentences in the corpora. For this reason, understanding the types of sentences we do not expect to parse lays the groundwork for understanding what the inferred grammars should parse, but don't. A number of lexical types that BASIL does not infer will prevent the grammar from having lexical coverage over sentences that contain those types of words. These include but are not limited to adjectives, adverbs and 'particles' marking complementation, subordination, information structure, questions and possession. Because these words may be homophonous with words that BASIL does handle, sentences with these lexical types may have lexical coverage and the grammar might even produce one or more parses for them, but those parses will not be correct. In addition, there are phenomena whose analysis doesn't depend on particular lexical items, but rather phrase structure rules for specific configurations (e.g. asyndetic coordination) or lexical rules for particular types of inflection (e.g. imperatives), or both in combination (e.g. adverbial clauses where subordination is marked morphologically). If the inferred grammars don't cover a phenomenon, we don't expect the grammars to parse sentences including that phenomenon (correctly, or at all).

Some parses have the correct predicate-argument structure but lack some semantic features as a result of out-of-scope syntactic phenomena that contribute information to the semantic structure. As an example, yes/no questions and imperatives are traditionally modeled in the DELPH-IN formalism with the SF (sentential force) feature, which can have the values prop (proposition), ques (question) or comm (command) (Flickinger et al., 2014b). The inferred grammars for some languages parse questions and imperatives with the correct predicate-argument structure, but they do not use the appropriate prop or comm, so the correct features are not fully specified. With this context established, the next subsection presents the performance of the development grammars.

5.3 Coverage for Dev Languages

We evaluated system performance on the development languages using 10-fold cross validation. We assessed the inferred grammars by parsing sentences in their respective test folds, using five metrics: *lexical coverage* — the proportion of sentences for which the grammar has an analysis for each word; *parse coverage* — the proportion of sentences for which the grammar can produce a syntactic analysis; *correct predicate-argument structure* — the proportion of sentences the grammar parses, producing a semantic representation that includes appropriate predications and arguments for each semantic entity; *correct predicate-argument structure and semantic features* — the proportion of sentences for which

Phenomenon	# possible	# targeted by inference	# inferred from dev languages
noun lexical entry	4	2	2
verb lexical entry	4	2	2
auxiliary lexical entry	6	4	4
adposition lexical entry	3	3	3
morphological rule	5	5	5
person	9	8	4
tense	2	1	1
word order	10	9	6
determiner order	4	4	4
auxiliary order	9	9	7
case system	9	3	2
argument optionality	18	15	12
sentential negation	41	23	9
coordination	12	11	10
total	136	99	71

Table 3: The number of possible values for the closed set features to define phenomena in the grammar specification and, those targeted by the inference system and those attested in the development languages

Feature Category	# Found
Number	4
Gender	5
Case	21
Tense	6
Aspect	16
Mood	14
Total	66

Table 4: The number of morpho-syntactic features found in the development languages. (Person features are not included because the Grammar Matrix defines them automatically based on the overarching person system.)

the grammar produces the correct predicate-argument structure as well as the appropriate PNG and TAM features on those arguments and the correct sentential force; and *ambiguity* — the average number of results per sentence that parses. For details on how we operationalized these metrics, see Section 6.

Table 5 presents the results using these metrics for each of the development languages. Whereas calculating the lexical coverage, parse coverage and ambiguity are automated processes, calculating the correct predicate-argument structure and features requires manual inspection of the semantic representations (for a detailed description of these processes, see §6.1). For this reason, we provide results for correct predicate-argument structure and correct predicate-argument structure and features across all folds for languages with less than 1,000 IGT, but for those with more IGT, we provide these metrics only for the first fold.

The sentences for which the grammar produces a semantic representation with the correct predicate-argument structure and features are a subset of those for which the grammar produces a semantic representation with the correct predicate-argument structure. In turn, those are a subset of the sentences with parse coverage, which are a subset of those with lexical coverage. This is illustrated by the bar graph in Figure 11.

To contextualize this performance, remember that the datasets come from a wide range of sources. Transcribed speech and elicitations often include sentence fragments, which the grammar will not accept as sentences. For this reason, and because of the many out-of-scope phenomena described above, we do not expect the inferred grammars to parse a very large portion of the held-out sentences they are tested on. Instead, the most useful comparison to consider is the number of sentences that parsed with the correct predicate-argument structure or correct predicate-argument structure and features versus the number of sentences that parsed, but did not have the correct semantic representation.

Previously, little work has been done that evaluates inferred grammars on held-out test items. Hellan (2010) and Hellan and Beermann (2011) do not present any evaluation for their inference system and Indurkhya (2020) evaluates his grammars over the same sentences as were seen in the training set. However, Bender et al. (2014) and Zamaraeva et al. (2019a) evaluate inferred grammars over held-out portions of the Chintang dataset. Here we use the same dataset of Chintang as one of our development sets, so we use Zamaraeva et al. 2019a as a point of external comparison.

By creating lexical items for determiners, adpositions, coordinators and negation words, we doubled the

Language [iso]	Lexical Coverage (%)	Parse Coverage (%)	Correct Pred-Arg Structure (%)	Correct Pred-Arg Strugure and Features (%)	Ambiguity
Abui [abz]	53.19	41.96	10.19*	5.73*	2195
Chintang [ctn]	22.29	12.24	3.58*	1.94*	5562
Haiki [yaq]	17.49	10.29	1.79*	0.89*	161
Lezgi [lez]	7.88	6.08	0.00*	0.00*	10419
Matsigenka [mcb]	12.61	8.02	1.15	1.15	2333
Meithei [mni]	5.86	5.24	1.05	0.42	3722
Nuuchahnulth [nuk]	23.09	10.14	1.87	1.09	265
Wambaya [wmb]	9.41	2.08	0.98	0.12	4
Tsova-Tush [bbl]	28.79	24.05	4.35*	0.00*	3418

Table 5: Coverage and Ambiguity for Development Languages. Results are averages across 10 folds. * indicates results for only a single fold

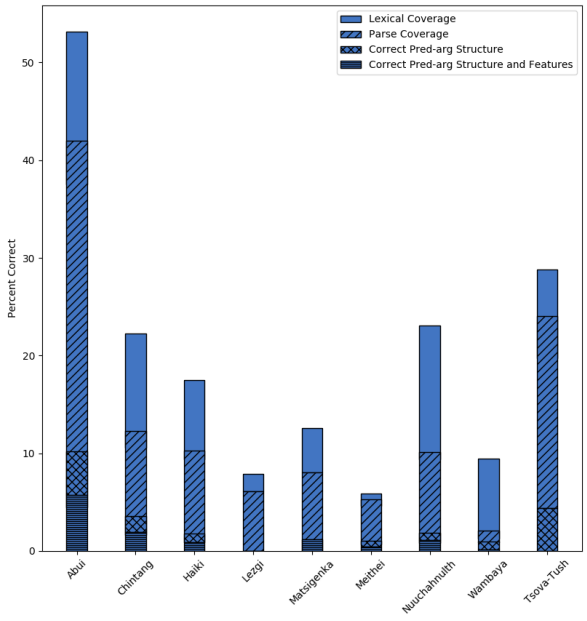


Figure 11: Lexical coverage, parse coverage, correct pred-arg structure and correct features by language for development languages

number of test items for which the inferred grammars can analyze each word, compared to Zamaraeva et al. (2019a) for Chintang. This is critical as the grammar has no chance at syntactic analysis if lexical analysis fails. Our lexical coverage averages 20% across the development languages. (Here and throughout, we use macro-averages weighting each language equally.)

The next thing to consider is what portion of the sentences for which the grammar can analyze each word can be analyzed syntactically. Zamaraeva et al.'s inferred Chintang grammar parsed 30% of the sentences it had lexical coverage for. Our inferred grammars have significantly closed that gap, parsing 84% of the Chintang sentences that had lexical coverage and 67% of the items with lexical coverage on average across all of the development languages.

The most important metric is the proportion of test items the grammar parses correctly. On the development languages, the number of sentences BASIL parses with the correct predicate-argument structure ranges from 0% to 10%. The number of sentences with correct predicate-argument structure for Chintang is more than double what it was for Zamaraeva et al. (2019a) and the introduction of semantic features increases the quality of these parses. BASIL has more spurious coverage than the system of Zamaraeva et al. (2019a), which correctly parsed 47% of its parsed sentences. BASIL produced parses with correct predicate-argument structure for only 19% of the Chintang sentences it parsed; however, for 9% of the sentences it parsed, BASIL also included the correct features in the semantic representation.

Finally, measuring ambiguity shows how many incorrect or redundant parses are produced by the grammar. Ideally, this should be minimal, as in Wambaya, for which our inferred grammars average four parses per sentence. However this average increases when there are multiple analyses for a morphological or syntactic phenomenon, some of which are valid and some of which are not. We go into this in more detail in Section 8.3 where we compare the ambiguity of the inferred grammars with baseline inference systems. At this stage, we simply note that there is an inherent trade-off between coverage and ambiguity in inferred grammars, just as in hand-crafted grammars: Where sentences may seem unambiguous to humans, who have the benefit of context and world knowledge, computers are much better at finding alternative, often pragmatically odd, analyses. The more phenomena a grammar includes, the more such analyses are available.

5.4 Summary

In this section we described the languages and datasets that we used during development and assessed BASIL in terms of how it performs on them. We primarily used 9 development languages from 7 language families, but at times consulted others for a total of 27 languages from 19 families, in order to make BASIL as robust to cross-linguistic variation as possible. We showed that the 9 development languages tested most of the phenomena targeted by the inference system and performed well in terms of producing grammars that handle those phenomena correctly. With this performance at the end of development, we turn to evaluation on held-out languages to determine how well BASIL generalizes to previously unconsidered languages.

6 Evaluation Methodology

In Section 5, we present results for our development languages, where system development benefited from close error analysis. We use the same methodology to evaluate the system on held-out data from held-out languages. As above, we use the full end-to-end pipeline described in Section 3, with 10-fold cross-validation, and report the same five metrics from Section 5.3: lexical coverage, parse coverage, correct predicate-argument structure, correct predicate-argument structure and semantic features, and ambiguity. In this section, we describe how we measured these (§6.1), and present our baseline system (§6.2) and test languages (§6.3). The following sections (§§7–8) present our results and error analysis on the held-out languages.

6.1 Evaluation Metrics: Parsing and Treebanking

After inferring a grammar from the training data, we use the ACE parsing software (Crysmann and Packard, 2012) to parse each sentence in the test dataset (links to ACE and other software used for evaluation can be found in Appendix B). For each sentence, ACE outputs whether the grammar had a lexical analysis for each word in the sentence, from which we calculate *lexical coverage*. If each word has an analysis and the grammar accepts the sentence as grammatical, ACE returns a result which includes the syntactic parse trees and corresponding semantic representations (illustrated in Figures 12 and 13), and on this basis, we calculate *parse coverage*. In many cases the grammar contains *ambiguity*, returning multiple parses per sentence, and we report this as the average number of results for sentences that parse.

The process of finding the *correct predicate argument-structure* (and *semantic features*) is more involved. After parsing the test sentences with ACE, we use the Full Forest Treebanking software (FFTB; Packard, 2015) to examine the lexical and syntactic rules in the parse forest to identify any trees that represent an appropriate syntactic parse for the sentence. We then inspect the corresponding semantic structure by looking at the predicate-argument structure as well as the semantic features on each argument. Consider the syntactic and semantic representations in Figures 12 and 13 which were produced by an inferred grammar for the Matsigenka sentence in (6).

(6) Ikamagutakerotyo.
i-kamagu-t-ak-i=ro=tyo
3mS-look-EPC-PERF-REALIS=3fO=AFFECT
'He looked at it.' [mcb] (Michael et al., 2013)

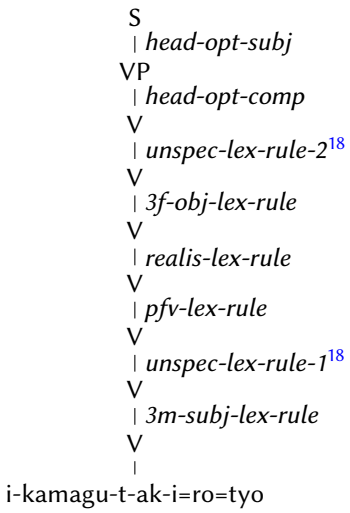


Figure 12: The syntax tree corresponding to the semantic representation in Figure 13

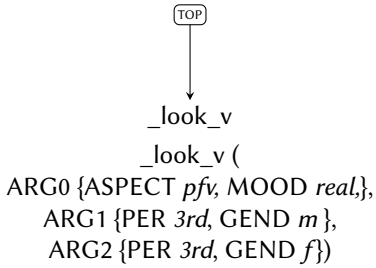


Figure 13: The best semantic representation produced by the inferred grammar for the sentence in (6)

[18]We use 'unspec' as a naming convention for lexical rules that do not add any morpho-syntactic or morpho-semantic features.

Sentence (6) has only one word[19] but includes three semantic arguments: an event and two entities. For this reason, the tree in Figure 12 contains a series of lexical rules (the nodes labeled as V) and two syntactic rules (object dropping, labeled by VP, and subject dropping, labeled by S).[20] The semantic dependency contains only one predicate, which is contributed by the verb *kamagu* 'look'. That predicate has three arguments. First is the event argument (ARG0), which is marked with perfective aspect and realis mood. Next there is the semantic argument (ARG1) corresponding to the unexpressed subject, which is marked with third person and masculine gender, and third is the semantic argument (ARG2) corresponding to the unexpressed object, marked with third person and feminine gender.

We consider the semantic representation in Figure 13 to have the correct predicate-argument structure because it contains all of the predications that should be in the semantic representation and no additional, incorrect predications, and because the predication has the correct arguments: an event and two entities. We consider the semantic features in Figure 13 to be correct because they reflect all of the semantic features that A) are in the IGT and B) the inference system targets: BASIL only targets PNG and TAM features, so those are the only ones we expect. The semantic representation does not reflect the affective meaning because BASIL does not extract stance features.[21]

Although using treebanking to check parses for correctness is an established practice (see inter alia Oepen et al., 2002; Flickinger et al., 2017), assessing the accuracy of semantic representations for languages that one doesn't speak fluently and isn't an expert on is a challenging task. For example, it can be hard to know if some locative dependents are core arguments of the verbs or if they are modifiers. Furthermore, glossing conventions vary from linguist to linguist and with limited familiarity with the datasets, one must make guesses as to implications of some grams and the ambiguous cases one might encounter are difficult to anticipate without first engaging with the data. Therefore, we established a practice of consulting both the gloss line and the translation line as the translation line might omit or add some semantic information compared to the gloss line, but the gloss line may be ambiguous with regards to which words are arguments of which and this can be learned from the translation.[22] After developing basic guidelines by discussing some specific examples from the development datasets, the authors of this paper independently treebanked one fold from each of the Abui and Chintang datasets. These folds contained approximately 100 parsed IGT each.

	Abui [abz]	Chintang [ctn]
Correct Parse	0.5714	0.7843
Matching Pred-Arg Structure	0.5714	0.7843
Matching Features	0.5714	0.5882
Exact Match MRS	0.5143	0.5882

Table 6: F1 scores for inter-annotator agreement on treebanked coverage for Abui and Chintang

Following the methodologies set forth by Dridan and Oepen (2011) for semantic evaluation and Bender et al. (2015) for inter-annotator agreement (IAA), with some adaptations to target our task-specific goals, we calculated IAA for the treebanked results of the two development sets, which we present in Table 6. Dridan and Oepen (2011) propose an Elementary Dependency Match (EDM) score calculated from multiple parts of the semantic representation. We used their EDM_{na} metric for naming and argument identification, and added a metric for semantic features. Following Bender et al. (2015), and in light of the lack of chance-corrected metrics for such structures, we assess IAA for these metrics by calculating the F1 score for these metrics between the two annotators. These F1 scores are shown in Table 6 as Matching Pred-Arg Structure and Matching Features. To situate these measures we also present F1 scores for IAA for whether the parses for the item were considered to include one that was correct (Correct Parse) and whether the two semantic representations matched exactly (Exact Match MRS).

The F1 score for correct parse is the same for matching predicate-argument structure, which shows that when we agreed that there was a parse with an acceptable predicate-argument structure, we also agreed on what that predicate-argument structure should be.[23] Disagreements were often due to one author interpreting something as a modifier instead of an argument (the inferred grammars do not handle modifiers, so these parses would be rejected) or whether sentence fragments should be accepted or rejected, given an otherwise correct semantic representation.

The slightly lower F1 for Exact Match MRS for Abui is due to a slightly different but equally acceptable

[19] Although Michael et al. use an = to indicate two clitics (*=ro* and *=tyo*), BASIL analyzes them as affixes. We made this analytical choice because = in IGT frequently indicates less phonologically integrated affixes, rather than clitics in the sense of Zwicky and Pullum (1983).

[20] The treatment of these arguments as a dropped subject and object is consistent with Inman's (2015) analysis of pronoun incorporation in Matsigenka.

[21] The gloss AFFECT is not explicitly defined by Michael (2008), but from his discussion around such examples, we believe that this refers to stance. We assume that EPC marks an epenthetic consonant, and does not contribute any semantic feature.

[22] This is based on Bender's previous treebanking work in Bender 2008a, Bender et al. 2014 and Zamaraeva et al. 2019a.

[23] This does not necessarily mean that we chose the same syntactic parse, as spurious ambiguity may result in multiple syntactic structures producing the same semantic representation.

predication for the verb in one sentence: leave.for_v_rel vs. leave.for-or-step_v_rel, where the second represents two possible meanings of the verb. For Chintang the feature agreement is lower than predicate-argument structure agreement. For this language the grammars have a great deal of ambiguity in the lexical rules. In many cases, it was not possible to find a parse that had all of the correct features, and we chose parses with different subsets of correct and incorrect features.

After discussing our disagreements, we extended our definitions of correct parses. For all held-out languages a single author treebanked the results, according to the conventions decided through this process.

6.2 Baseline

The primary contribution of this paper is in inferring syntactic properties from IGT data and integrating these with lexical and morphological properties inferred by MOM (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017). Therefore we compare our results to three baseline systems that are morphologically and lexically robust with respect to accounting for the training data, but are syntactically naive. Each of these use lexical entries and morphological rules from MOM for nouns and verbs. Although MOM extracts morpho-syntactic features for nouns and verbs and adds them to the lexicon and morphological rules, inference is required to define them appropriately in the grammar specification. Because a grammar specification with morpho-syntactic features on verbs and lexical entries with no definition of those features would not result in a working grammar we disable the feature extraction in MOM for all baselines.

Table 7 enumerates the syntactic specifications for our baseline systems. The first baseline (BROAD-COV) posits the specifications for each syntactic phenomenon we account for that we expect to result in the broadest coverage, given no specific knowledge of the language. The second baseline (TYP) posits the specifications that are typologically most common, according to the information available in WALS (Dryer and Haspelmath, 2013) and other typological resources. If a typologically-most-frequent choice could not be made, we select the specification at random if it is required by the Grammar Matrix, and omit it otherwise. Aside from specifications made at random (which are chosen with each run), the syntactic specifications under the BROAD-COV and TYP baselines are the same for all grammars, that is, they do not vary in response to the data presented. Finally the third baseline (RAND) selects a value for each specification at random. The baseline systems make a different random choice for each RC specification every time they are run, therefore the values in the baseline files for each fold of training data are different.

	BROAD-COV	TYP	RAND
word order	free	sov	RC
has determiners	yes	yes	RC
noun-det order	RC	noun-det	RC
det required	optional	RC	RC
has auxiliaries	no	no	no
verb valence	trans	RC	RC
case frame	none	none	none
s coordination	asyndeton		RC
vp coordination	asyndeton		RC
np coordination	asyndeton		RC
n coordination	asyndeton		RC
subj-drop	all	all	RC
obj-drop	all		RC

Table 7: Grammar specifications for syntactic phenomena for three baseline systems. RC indicates a random choice

Language	ISO 639-3	Source	Number of of IGT	POS tags in source
Arapaho	arp	Toolbox	5000	yes
Hixkaryana	hix	Toolbox	5749	yes
South Efate	erk	Toolbox	1875	yes
Titan	ttv	Toolbox	1799	yes
Wakhi	wbl	FLEx	683	yes

Table 8: Source, number of IGT and presence of POS tags for the held-out datasets

6.3 Held-out Languages

To test how well BASIL generalizes to new languages, we acquired datasets for five additional languages, which we did not consider during development and which are genealogically and geographically varied from the development languages. These languages are listed in Table 8 and the locations where they are spoken are shown in green on the map in Figure 10.

We pre-processed each dataset by filtering out ungrammatical examples (examples marked with a *) and removing duplicates. For held-out evaluation, we selected only languages with POS tags in the original dataset. This information as well as the type of source dataset and the number of IGT after filtering are summarized in Figure 8. In this section, we provide a brief description of each language and dataset. For a full list of citations for datasets and descriptive resources referenced in this section, see Appendix C.

Arapaho [arp] is an Algonquian language of the Algic language family with only about 250 native speakers in the United States (Cowell and Moss Sr, 2011). The dataset we use is a 5,000 item subset of a ~60,000 IGT corpus (Cowell, 2018), randomly selected from fully-glossed examples. The corpus includes elicitations and transcribed conversations, among other genres.

Hixkaryana [hix] is a Cariban language in the Waiwai

subgroup with about 1,200 speakers (Eberhard et al., 2019). After removing IGT with incomplete glosses, the corpus (Meira, 2020) contains almost 6,000 IGT.

South Efate [erk] is a Vanuatu language of the Austronesian language family, spoken by about 6,000 people on the Efate island in the Republic of Vanuatu (Thieberger, 2006b). From the 3,000 IGT corpus (Thieberger, 2006a), we use 1,900 fully glossed examples.

Titan [ttv] is also an Austronesian language, and while it and South Efate are both Oceanic, Titan is grouped as a language of the Admiralty Islands while South Efate is Central-Eastern Oceanic. The various dialects of Titan are spoken by approximately 3,500-4,500 people (Bowern, 2011). This corpus contains just under 1,800 IGT after filtering for glossing (Bowern, 2019). For this corpus, we obtain POS tags from the accompanying Toolbox lexicon. This introduces some noise, due to lexical ambiguity, but less than if we had used the projected POS tags from INTENT.

Wakhi [wbl] is an Iranian language of the Indo-European language family and is spoken primarily in Afghanistan and has a growing speaker population of about 17,000 (Eberhard et al., 2019). The dataset is small, containing only about 700 IGT after filtering (Kaufman et al., 2020). However, it is thoroughly glossed and is made up primarily of elicitations targeting specific syntactic phenomena.

7 Results

Using the methodology in Section 6, we performed ten-fold cross-validation on the evaluation languages for the BASIL inference system and the three baselines described in Section 6.2.[24] We show lexical coverage in Table 10, parse coverage in Table 11, coverage with correct predicate-argument structure in Table 12, coverage with correct predicate-argument structure and semantic features in Table 13 and ambiguity in Table 14.

For each language, we treebanked *n* folds such that the number of parsed sentences in *n* folds is greater than 100. The results for lexical coverage, parse coverage and ambiguity are averages across ten folds, while the results for coverage with correct predicate-argument structure and coverage with correct predicate-argument structure and features are averages across *n* folds where *n* is given in Table 9.

There is a great deal of variation in how well any of the systems did at inferring grammars that can parse held-out sentences for each language, as illustrated by the graph in Figure 14. Coverage for Arapaho was very low, at roughly 3% lexical coverage for each system and similar parse coverage for BASIL and BROAD-COV. Across all systems, Hixkaryana and Wakhi had significantly higher lexical and parse coverage, exceeding BASIL's performance on most of the development languages. South Efate and Titan fall between these two extremes. The correct coverage is more consistent across languages with Wakhi as an outlier. For Wakhi, BASIL achieves correct predicate-argument structure for 14.20% of the items in the test set and correct predicate-argument structure and features for 5.8% and the BROAD-COV baseline achieves 12.75% correct predicate-argument structure, while the remaining languages have much lower correct coverage across systems. Finally, the ambiguity (or average number of parses per parsed item) for these languages is quite low for Wakhi, on the order of tens, and extremely high for South Efate, on the order of 100,000. We provide more detail on the causes of ambiguity in the inferred South Efate grammar in Section 8.3.

Language	Treebanked folds (*n*)	Parsed sentences in *n* folds	Total sentences in *n* folds
Arapaho [arp]	7	109	3500
Hixkaryana [hix]	1	198	575
South Efate [erk]	7	110	1504
Titan [ttv]	6	110	1080
Wakhi [wbl]	5	115	345

Table 9: Number of sentences treebanked across *n* folds for each held-out language

Overall, the systems performed best on Wakhi across the five metrics. Performance for Hixkaryana, South Efate and Titan was somewhat lower, with coverage for Arapaho being the lowest. In Sections 8.1 and 8.2, we explore sources of this variation, including characteristics of the languages and of the IGT datsets.

To understand the impact of *syntactic inference* on automatic grammar generation, we compare BASIL with three baselines that use the same morphotactic and lexical inference system as BASIL, but must specify the syntactic portions of the grammar specification through some other means. The BROAD-COV system uses the specifications that are expected to parse the most sentences, whether correctly or incorrectly. TYP uses the typologically most common specification and RAND uses a random choice (for details, see §6.2). Each of these baselines uses a random choice for at least one specification, where no clear determination could be made for broad coverage or typological frequency, so ten-fold cross validation (given that a new random choice is made when specifying the grammar for each fold) is important to reduce the effect of chance on the overall performance of each baseline.

Because the same morphotactic and lexical inference system was used for the baselines as for BASIL, the lexical coverage across systems is roughly compa-

[24] The code to reproduce these results is available at https://git.ling.washington.edu/agg/repro/basil-2020.

Language	BASIL	BROAD-COV	TYP	RAND
Arapaho [arp]	3.64	3.52	3.64	3.18
Hixkaryana [hix]	38.09	36.01	35.88	35.92
South Efate [erk]	12.80	13.55	14.29	13.17
Titan [ttv]	13.56	19.40	20.34	19.40
Wakhi [wbl]	39.68	29.72	31.48	31.04

Table 10: Lexical coverage for held-out languages as a percentage of the total number of test items across ten folds

Language	BASIL	BROAD-COV	TYP	RAND
Arapaho [arp]	3.04	3.06	0.50	0.26
Hixkaryana [hix]	34.18	31.28	2.80	1.25
South Efate [erk]	6.77	9.81	0.27	0.27
Titan [ttv]	10.34	16.18	0.06	0.17
Wakhi [wbl]	30.31	24.89	10.25	3.22

Table 11: Parse coverage for held-out languages as a percentage of the total number of test items across ten folds

Language	BASIL	BROAD-COV	TYP	RAND
Arapaho [arp]	0.17	0.20	0.00	0.03
Hixkaryana [hix]	2.26	2.26	1.57	0.52
South Efate [erk]	0.38	0.31	0.00	0.00
Titan [ttv][25]	0.28	0.65	0.09	0.19
Wakhi [wbl]	14.20	12.75	2.61	0.58

Table 12: Coverage with correct predicate-argument structure as a percentage of the total number of test items across *n* folds

Language	BASIL	BROAD-COV	TYP	RAND
Arapaho [arp]	0.09	0.06	0.00	0.00
Hixkaryana [hix]	0.00	0.00	0.00	0.00
South Efate [erk]	0.15	0.00	0.00	0.00
Titan [ttv]	0.19	0.00	0.00	0.00
Wakhi [wbl]	5.80	0.58	0.00	0.00

Table 13: Coverage with correct predicate-argument structure and semantic features as a percentage of the total number of test items across *n* folds

Language	BASIL	BROAD-COV	TYP	RAND
Arapaho [arp]	145	936	4	3
Hixkaryana [hix]	5642	15596	2	6
South Efate [erk]	126379	9759	2	4
Titan [ttv]	595	6201	2	1
Wakhi [wbl]	10	26	1	2.5

Table 14: Average number of results per parsed sentence for across ten folds

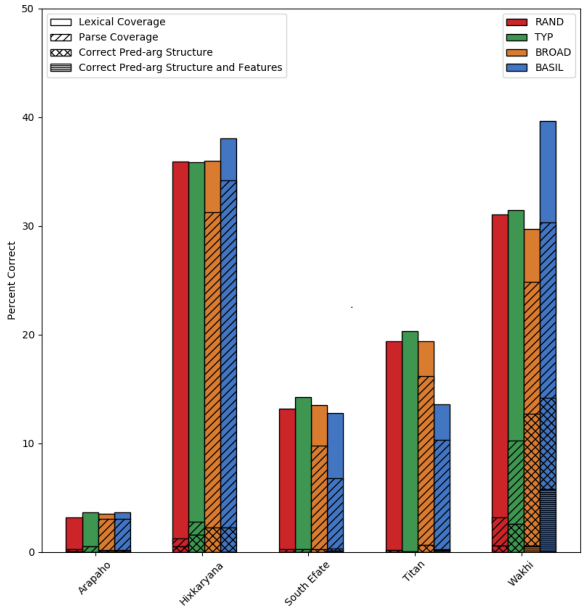


Figure 14: Lexical coverage, parse coverage, correct pred-arg structure and correct features by language for held-out languages

rable. For some languages, the baseline lexical coverage is lower because the baselines can only use POS tags to identify lexical items, while BASIL uses additional heuristics. For other languages, it is slightly higher because BASIL strategically excludes ditransitive and clausal complement-taking verbs (which it would not handle correctly) from the lexicon.[26] Additional variation in the lexical coverage across systems can be attributed to variations in the morphological graph: It is different for each baseline, because it is sensitive to verb valence assignments and these are done at random in each run for the TYP and RAND baselines.

A larger and more meaningful difference between the systems is seen in parse coverage. Here, the TYP and RAND baselines have much lower coverage than BASIL and BROAD-COV. While the TYP baseline has a better chance of using the correct value for each individual specification, it will not necessarily be correct for enough phenomena to produce a grammar that can parse simple sentences: For example, even if the order of verbs with respect to subjects and objects is correct, sentences with determiners won't parse if the determiner-noun order is incorrect. By design, the BROAD-COV system has the highest parse coverage, often outperforming BASIL; however, without syntactic in-

[25] For Titan we report a correct coverage that is higher than the parse coverage for the TYP and RAND baselines. This is possible because there were more parsed items per fold in the 6 folds we treebanked than in the remaining 4.

[26] BASIL cannot properly account for ditransitives as they are not currently supported by the Grammar Matrix. Clausal complement-taking verbs have also been left out of scope at this time.

ference this coverage could be spurious, so we must consider correct coverage (described in §6.1). Again, the TYP and RAND baselines under-perform the other systems, as there is a relatively low chance that their specifications will correctly model any given language. In terms of correct predicate-argument structure, BASIL outperforms BROAD-COV for South Efate and Wakhi, while BROAD-COV does better for Arapaho and Titan. They tie on Hixkaryana. As BROAD-COV is designed to maximize coverage, it specifies asyndetic coordination for each language, enabling it to parse sentences for languages where BASIL failed to infer this strategy. For correct predicate-argument structure and semantic features, BASIL outperforms all baselines, as they cannot posit semantic features. Only in rare cases did BROAD-COV have the 'correct features', because the semantic representation shouldn't include any features at all.

So far, we have shown that BASIL and BROAD-COV out-perform the other two baselines in parse coverage and correct predicate-argument structure, while BASIL out-performs all of the baselines in correct predicate-argument structure and semantic features, as illustrated in Figure 14. The last thing to consider is how much ambiguity each of the grammars contain. TYP and RAND produced grammars with very little ambiguity. These grammars only parsed simple sentences, so low ambiguity is not surprising. BROAD-COV was designed to maximize coverage, but this comes at the cost of increased ambiguity. For example, positing free word order for each language will ensure that all word orders will parse, but will also allow parses where the wrong constituents are identified as subjects and objects. As a result, the BROAD-COV baseline has significantly higher ambiguity than BASIL for all languages but South Efate.

While the results show a great deal of variation across the test languages, BASIL and BROAD-COV outperform the TYP and RAND baselines for most metrics. BASIL and BROAD-COV perform fairly comparably for a number of the metrics, but BASIL excels in two areas. First, BASIL generally has fewer parses per test item than BROAD-COV, suggesting that there is less spurious ambiguity in the inferred grammars than in that baseline. While TYP and RAND have even lower ambiguity scores, they also have such low coverage that this is not an advantage. Second, the semantic representations produced by BASIL are more correct in that they contain semantic features, resulting in higher scores for the correct predicate-argument structure and features metric.

8 Error Analysis

8.1 Out of Scope Phenomena

We begin our error analysis by establishing first what we do not expect BASIL's grammars to parse. Focusing on sentences where lexical coverage was achieved but the sentence did not parse or parsed incorrectly, we describe phenomena that are frequent in the test data but are beyond the scope of the current inference system.

BASIL currently handles a number of lexical types such as transitive and intransitive verbs, auxiliaries, nouns, determiners and case-marking adpositions, as well as phenomena including word order, case, argument optionality, sentential negation and coordination. However, it does not yet handle a number of very common phenomena such as adjectives, adverbs, ditransitive or clausal complement-taking verbs, content question words, possessives, etc. Therefore, sentences containing these lexical items will only have lexical coverage if a lexical item was inferred in error. At the same time, sentences that contain these syntactic phenomena will not parse at all or will not parse correctly.

In particular, frequent error types include: (i) verb valence, where BASIL posited intransitive or transitive entries for verbs which were actually ditransitive or clausal-complement taking; (ii) adnominal possession, where grammars produced by BASIL parsed but could not attribute the correct semantics to examples with possession; (iii) vocatives analyzed as subjects or objects; (iv) sentence linkers parsed as coordination; and (v) disfluency markers (e.g. *P* for 'pause') analyzed as verbs.

8.2 In Scope Phenomena

Whereas the previous section described common errors due to out of scope phenomena in the test data, this section focuses on errors due to BASIL failing to correctly infer phenomena that it was designed to handle. The sources of these errors range from the input data to problems with BASIL's inference algorithms or their implementation.

8.2.1 Wrong Part-of-Speech

Both BASIL and MOM rely on POS tags in the input to identify nouns and verbs. In some cases, the POS tag in the corpus may be incorrect. For example, in (7) the word *titko* is glossed as 'brazil.nut' but marked with a verbal POS tag. Such errors are not uncommon, as even the most careful human annotation is subject to error.

(7) Tutko titko Brazil.nut Vt yakahetxkoni. y-akaha-yatxkoni REL-break-DPST2:COL prs-Vt-tamn
'They were shelling Brazil nuts.' [hix] (adapted from Meira, 2020)

Because *titko* is glossed as a verb, the inferred grammar treats it semantically as an event instead of as a

participant of the breaking/shelling event, resulting in an incorrect semantic representation.

8.2.2 Wrong Predication

We considered it an error anytime the predication associated with a word did not reflect the meaning in the gloss, even if the overall shape of the predicate-argument structure was correct. This can occur if MOM's heuristics for locating the root of a word fail in a particular case. For example, the IGT in (8) had spaces on both sides of the second hyphen. MOM guessed that the hyphen belonged to *neeni*, which in turn meant that *t* was the root, leading to a lexical entry with the predication _3.S_v_rel.

(8) Nehe' hinen nihneenit.
nehe' hinen nih- neeni - t
this man PAST- itis - 3.S
'The man was the one.' [arp] (adapted from Cowell, 2018)

8.2.3 Missed Semantic Features

BASIL's greatest advantage over the baseline systems is its addition of semantic features to the grammars, but it still made some errors in feature inference. There is significant variation in the way linguists gloss syntactico-semantic features, and BASIL's most straight-forward source of error for semantic features was in not properly identifying all grams in the held-out corpora. BASIL uses a large dictionary of glosses, which it maps to 116 common PNG, TAM and case grams to identify morpho-syntactic and morpho-semantic features (see §4.1). Even so, the held-out corpora included grams that were not in this dictionary. In particular, this dictionary did not include any glosses for the pluperfect aspect 'PLPF', which appears in Wakhi, the immediate past 'IPST' or distant past 'DPST' used in Hixkaryana, or the narrative past 'NARRPAST' used in Arapaho. In addition, while the dictionary included 'D' as a gloss for dual number and quite a few person and number combinations (e.g. '3DU'), it did not contain '3D' which is used for third person, dual number in the South Efate corpus. This led to test items, which otherwise parsed correctly, not including all of the semantic features.

8.2.4 Auxiliaries

BASIL treats words that have only TAM and/or PNG agreement features as auxiliaries (see §4.5.1). The abundance of TAM auxiliaries in the held-out languages, such as the future tense auxiliary in (9), revealed a bug in our implementation of auxiliary inference. The clause in BASIL's code that infers where the auxiliary occurs (before or after its complement) assigns the wrong value. This caused some inferred grammars to require auxiliaries after their verbal complements instead of before. Though our development languages included auxiliaries, these freer word order languages (Wambaya and Nuuchahnulth) did not reveal this bug.

(9) Tumrə maz̥ jittu.
tumrə maz̥ jaw-tu
FUT 1SG.OBL eat-PLPF
'I will have had eaten.' [wbl] (adapted from Kaufman et al., 2020)

8.2.5 Coordination

Coordination inference, described in Section 4.5.5, errs on the side of positing VP coordination unless it finds explicit evidence of S coordination in the form of a projected subject dependency that intervenes between the coordinator and a verb in the coordinand. This algorithm may be too aggressive because dependency tag projection is not always successful. In addition to that, the algorithm does not consider cases where the subject is dropped or cases where there is no coordinator, because an asyndetic strategy is employed. Because the inference of S coordination relies on an overt coordinator, sentences like the one in (10) from Titan are taken by BASIL as evidence of VP coordination instead of S even though each coordinand has an overt subject. Thus asyndetic S coordination isn't added to the grammar and examples like this can't be parsed.

(10) I ani pou i ani ma.
i ani pou i ani ma
3SG eat pig 3SG eat taro
'He ate the pig and he ate the taro.' [ttv] (adapted from Bowern, 2019)

In addition, examples of monosyndetic S coordination in Wakhi were misclassified as VP coordination because of failure to align the subjects between the English translation and the sentence. This prevented BASIL from inferring S coordination strategies and adding them to the grammar specifications. Because the BROAD-COV baseline posits asyndetic S coordination for all languages, that baseline was able to correctly parse sentences with asyndetic S coordination in Titan and Wakhi, giving it a boost in coverage over BASIL.

8.2.6 Case Frame

Finally, BASIL relies on the overt case markings on the subject and object (according to projected dependencies), to account for quirky case (§4.5.2). However, if no overt argument is found, the verb's case frame remains under-specified until it is merged with another instance

of the same verb. Even though BASIL inferred the overarching nominative-accusative pattern for Wakhi, it found verbs in the training data with oblique subjects which were merged with verbs that did not have overt case marking on their subjects. Because of this, the inferred grammars for some of the Wakhi folds included a rather large transitive verb class with oblique case on the subject, resulting in a number of IGT with overtly marked nominative subjects in the test data that did not parse.

8.2.7 Summary

The majority of errors discussed in this section come from lexical inference. Beyond that, we identified three main sources of error in the syntactic specifications. One was a bug that resulted in auxiliaries having the wrong order with respect to their complements. Resolving this bug is trivial, while the errors in S coordination and case-frame inference require some re-designing of the algorithms. In particular, BASIL requires too much evidence to infer S coordination. As future work, we propose modifying the algorithm to rely less on projected dependencies and instead to leverage the dependency parse of the English translation to distinguish between VP and S coordination in the translation. The same redesign could be applied to N and NP coordination as well. The case frame inference algorithm may assign quirky case too readily and rather than merging lexical items with no case frame with those that have quirky case, should assign default case to those verbs unless a verb with the same orthography is found with quirky case in the corpus. Alternatively, better verb classes could be inferred with some re-tooling of the interaction between BASIL and MOM, so that case frame inference happens after morphotactic inference, similar to the pronoun and auxiliary inference methodologies in Section 4.2.2.

8.3 Ambiguity

BASIL's inferred grammars generally had less ambiguity than the BROAD-COV baseline for two intuitive reasons. First, the free word order, argument optionality and coordination specifications in BROAD-COV introduce a lot of ambiguity in the number of ways nouns and verbs can combine. Second, BASIL's specifications for case frame and agreement further constrain which arguments can be subjects and objects, even in freer word order languages. In spite of this, BASIL's grammars for South Efate have significantly more ambiguity than BROAD-COV's. To shed light on this, we present a specific example from the fourth test fold from South Efate.

First of all, BASIL infers free word order, subject and object dropping and asyndetic coordination for VPs and NPs for this fold. Because of this, BASIL's inferred grammar is not less ambiguous than BROAD-COV in those areas. In order to understand why BASIL's grammar is even more ambiguous than BROAD-COV's, we explore the parse forest for the sentence in (11), which has asyndetic coordination, lexical ambiguity, morphological ambiguity and no overt case marking.

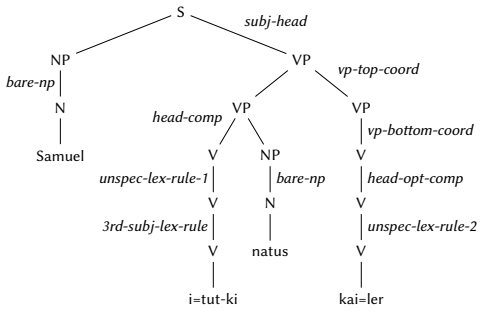


Figure 15: The parse tree generated by the BASIL and BROAD-COV grammars that corresponds with the semantic representation in Figure 16 for the sentence in (11)

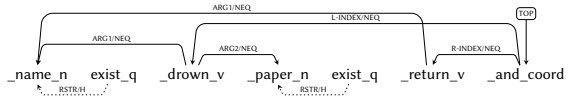


Figure 16: The best semantic representation generated by the BASIL and BROAD-COV grammars for the sentence in (11)

For this sentence, BASIL's grammar produces 2448 trees, while BROAD-COV's produces 19.[27] The best reading, produced by both grammars, is shown in the parse tree in Figure 15 and semantic representation in Figure 16.

(11) Samuel itutki natus kailer.
Samuel i=tut-ki natus kai=ler
Samuel 3S.RS1-drown-TR paper ES1-return
'Samuel threw in the paper and went back.' [erk]
(Thieberger, 2006a)

We use the Full Forrest Treebanking software (FFTB; Packard, 2015) to efficiently investigate such large parse forests with discriminant-based tree selection (Carter, 1997). Figure 17 shows the choices among discriminants that we used to single out the tree in Figure 15 from the other 2447 trees in the parse forest.

The discriminants in Figure 17 are not ordered, and represent one of many paths in the decision space. The bottom 4 choices in the decision tree result in no difference in the semantic representation, yet combined they increase the ambiguity by a factor of 16. The *no-drop-lex-rule* is added by the Grammar Matrix's argument

[27] These numbers are estimates provided by FFTB based on the packed forest, as opposed to ACE, which we used for Table 14.

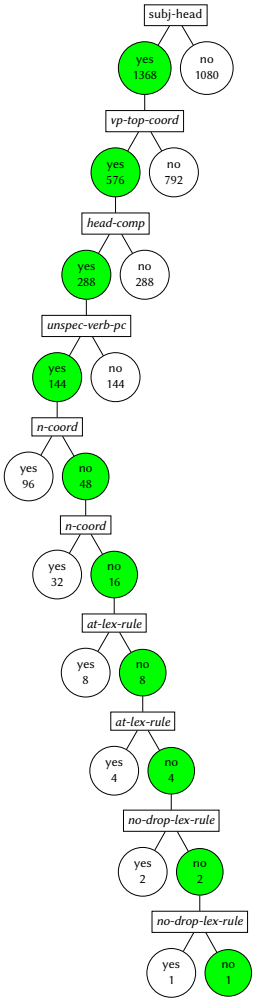


Figure 17: A decision tree illustrating the syntactic and lexical rules that discriminate between different parse trees produced by BASIL's grammar for the sentence in (11). The path in green shows the rules that we selected or excluded to identify the parse tree shown in Figure 15

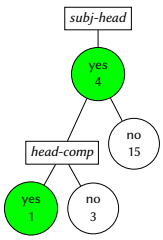


Figure 18: A decision tree illustrating the syntactic and lexical rules that discriminate between different parse trees produced by the BROAD-COV grammar for (11)

optionality library (Saleem, 2010; Saleem and Bender, 2010). This rule is intended to be further constrained by agreement restrictions for dropped arguments, but because BASIL does not add this information to the grammar, these optional, non-inflecting lexical rules add ambiguity for both verbs in (11). The two *at-lex-rule*s are added by the case library (Drellishak, 2009) for languages with case-marking adpositions. These rules apply to both nouns in (11) and because they apply optionally, each of these lexical rules and each of the words they apply to double the number of trees in the forest.[28]

In addition to these sources of ambiguity, there is an under-constrained noun coordination rule that applies optionally to each noun and can apply either before or after the bare-np rule, tripling the number of parse trees for each noun it can apply to. Because neither noun has an adjacent noun to attach to, these parses should not succeed, but they do as the result of a bug in the Grammar Matrix customization system.

All together the spurious case, coordination and argument optionality rules increase the number of possible trees by a factor of 144. Setting those aside, the number of possible trees looks much more reasonable. Additional ambiguity is added by two homophonous lexical rules for the *kai-* prefix: one adds first person agreement to the subject and the other (which produces the correct tree) does not add any features.[29]

The three choices at the top of the decision tree discriminate between trees in which *natus* is the object of *i=tut-ki* or *kai=ler* and indirectly, prevent *kai=ler* from being analyzed as a noun, coordinated with *natus*.

The decision tree for BROAD-COV to produce the parse shown in Figure 15 is shown in Figure 18. The lexical rules in the last four nodes in the tree in Figure 17 are not in the BROAD-COV grammar and therefore do not apply. Because ambiguity is a matter of combinatorics, the spurious lexical rules in BASIL's grammar inflate the ambiguity significantly. The same could be said for the sources of ambiguity in the BROAD-COV grammars for the other languages, where BASIL had less ambiguity.

Many of the sources of ambiguity in the South Efate grammars trace back to bugs in the Grammar Matrix customization system, rather than BASIL's inference. Furthermore, the high ambiguity for South Efate grammars was an outlier among the ambiguity in BASIL's grammars for the evaluation languages. This suggests that these sources of ambiguity, both from Matrix bugs and otherwise, are not particularly pervasive.

[28] The optionality of a non-inflecting lexical rule was a bug in the Grammar Matrix, and has since been addressed by (Conrad, 2021).

[29] The morpheme is glossed by the linguist as ES1. Thieberger (2006b) defines the ES abbreviation as "echo subject", and we assume that the 1 is a particular echo subject marker, but does not indicate first person, as there is no first person noun in the translation.

9 Conclusion

In this paper, we introduced BASIL — Building Analyses from Syntactic Inference in Local languages — a system for the automatic inference and generation of machine-readable grammars from IGT data. Leveraging the rich annotation in interlinear glossed text and syntactic information projected from parses of the English translation onto sentences in a local language, BASIL infers grammar specifications. These, in turn, can be input into the Grammar Matrix customization system to produce HPSG grammars.

BASIL utilizes an end-to-end pipeline that begins with an IGT corpus of a language and produces an HPSG grammar which can be loaded into parsing software to produce syntactic and semantic representations for strings in that language. Drawing on the linguistic information encoded in IGT text and generalizations about language from the typological literature, we designed algorithms that infer lexical and syntactic properties about a language and define these properties in a grammar specification. This grammar specification can be input into a grammar customization toolkit (the Grammar Matrix; Bender et al., 2002, 2010; Zamaraeva et al., forthcoming) to produce a machine-readable HPSG grammar for that language.

We built on previous work in grammar inference that produced both morphological (Wax, 2014; Zamaraeva, 2016; Zamaraeva et al., 2017) and syntactic (Bender et al., 2013, 2014; Howell et al., 2017; Zamaraeva et al., 2019a), specifications for a language. That work focused on lexical and morphotactic specifications for nouns and verbs, word order, case system and case frame for verbs. We integrated the existing modules into a single system which we scaled by adding inference for determiners, auxiliaries, case-marking adpositions, PNG and TAM features, argument optionality, negation and coordination.

The result is an inference system that identifies the overarching typological patterns for each of these phenomena and encodes that information in a grammar specification, which is then used to produce a grammar. As one of the goals of this work is to automatically infer grammars for a broad range of local and endangered languages, we developed inference algorithms using a data-driven process, testing our system on a genealogically and geographically diverse set of languages. During development, we consulted 27 languages from 19 language families, spread over 6 continents. We did end-to-end system testing on 9 of those 27 development languages.

In order to test the cross-linguistic generalizability of our inference system, we evaluated it using 5 languages from 4 language families that were not considered during development and did not come from any of the language families that we used in previous end-to-end testing. These languages were Arapaho, Hixkaryana, South Efate, Titan and Wakhi. We compared the performance of BASIL's inferred grammars with three baselines. The TYP baseline used the cross-linguistically most common specifications for each phenomenon (based on typological surveys), while RAND used random specifications. The low coverage of these baselines demonstrated that in order to produce a useful grammar, it is not sufficient to guess the right specifications for just some phenomena, but the specifications for a variety of interacting phenomena must be correct. The third baseline, BROAD-COV, was designed to parse as many sentences as possible in a language, and in spite of this, BASIL's overall coverage was comparable to BROAD-COV, while its grammars had less ambiguity for four of the five languages.

In addition to BASIL's parse coverage being higher than the TYP and RAND baselines and comparable with BROAD-COV, the semantic representations produced by BASIL's grammars were richer. In evaluation, we assessed not only the number of sentences that parsed, but the correctness of those parses in terms of the meaningfulness of their predications and the correctness of the argument relations for those predications. In this respect, BASIL and BROAD-COV performed comparably, outperforming the other two baselines by a large margin. However, BASIL's grammars also added semantic features for person, number, gender, tense, aspect and mood on the semantic predicates, resulting in even more detailed representations than those produced by the BROAD-COV grammars.

Because BASIL relies on the Grammar Matrix's typologically robust syntactic analyses to produce the grammars, BASIL can in principle be extended to account for phenomena as they are added to the Grammar Matrix. Recent work has added libraries for clausal complements (Zamaraeva et al., 2019b), adverbial clausal modifiers (Howell and Zamaraeva, 2018), nominalized clauses (Howell et al., 2018), adnominal possession (Nielsen, 2018; Nielsen and Bender, 2018) and constituent questions (Zamaraeva, 2021). Leveraging the analyses for these phenomena as well as others previously implemented in the Grammar Matrix, modules can be added to extend BASIL's scope.

Accounting for the characteristics of languages or datasets that have the most impact on system performance would enable better assessment of the system's weaknesses and ways to improve it. For this reason, we propose future work that systematically tests these factors by testing with different subsets of a single dataset with different sizes, genres, completeness of glossing or presence of part of speech tags. Upon identifying a threshold for these factors above which system performance stabilizes, it would then be possible to do more rigorous cross-linguistic testing to find language fami-

lies or typological properties that BASIL struggles with.

Acknowledging that BASIL's grammars are currently limited to a certain number of phenomena and are subject to some degree of error, we turn to a brief discussion of possible uses for these grammars both now and after additional inference modules are added. The first of these is in accelerating the process of creating machine-readable grammars, as creating grammar specifications, especially for languages with complex morphology, can be quite tedious.

Machine readable grammars that are somewhat larger than those produced by BASIL have been used for a broad range of applications such as data exploration (Letcher and Baldwin, 2013; Bouma et al., 2015), grammar checkers (da Costa et al., 2016) and automatic tutors (Hellan et al., 2013). Accelerating the process of developing this type of grammar increases the number of grammars that can be used for these applications. At the current stage, inferred grammars could still be useful for data exploration as they can be used to search corpora for the phenomena they model. This type of data exploration could assist linguists in finding relevant examples of specific phenomena they wish to analyze (as in Zamaraeva et al. 2017), or it could be used to help teachers find varied examples to use in lessons. Once a sufficient number of phenomena are handled by grammar inference, machine-readable grammars inferred from descriptive grammars could accompany those descriptive resources as a tool for further investigating the language's syntax, as described by Bender et al. (2012) and Bouma et al. (2015). Our inferred grammars for Wambaya, which were based on IGT extracted from Nordlinger 1998, serve as proof of concept for this possibility. Finally, as inferred grammars help to streamline the process of grammar engineering, ultimately grammars that started with BASIL and were extended by hand could be used to produce grammar checkers along the lines of da Costa et al. 2016 and other educational tools in order to assist in the effort of language revitalization.

Finally, there is potential for a symbiotic relationship between BASIL and typological resources such as WALS (Dryer and Haspelmath, 2013), SAILS (Muysken et al., 2016) and others. In particular, previous work has found that a number of the Grammar Matrix's specifications map directly to WALS features (de Almeida et al., 2019). For languages where these features are encoded in WALS, this information can potentially be incorporated into the grammar inference pipeline to improve the accuracy of inference for some phenomena. On the other hand, for languages whose features have not been added to databases like WALS, BASIL could be used to automatically infer those features, if an IGT corpus (or a descriptive grammar from which IGT can be extracted) is available.

The primary contribution of this work is a grammar inference system that takes an IGT corpus as input and produces a machine-readable, HPSG grammar that can be used for parsing and generation. Although previous work has automatically generated grammars for English and other languages frequently studied in NLP contexts, BASIL focuses on producing language technology in the form of syntactically precise grammars for local and endangered languages. In light of this, we tested the system on a large number of genealogically and geographically diverse languages and verified its cross-linguistic generalizability. Although the grammars produced by BASIL are still relatively low-coverage over corpora containing the complexity and variety inherent to human language, they provide a valuable starting point for producing broader coverage grammars which can be used to assist data exploration and language documentation and revitalization.

Acknowledgements

This work builds on the contributions of many researchers who have been involved in the AGGREGATION Project over the years. In particular, we would like to thank Olga Zamaraeva for her many contributions to MOM, including multiple changes and updates to the system which this work relies on centrally, Fei Xia for conceptual discussions, and Angelina McMillan-Major and previous AGGREGATION RAs who downsampled the Matsigenka corpus. In addition, Alexis Palmer, Kristy K. Phillips and Olga Zamaraeva developed a tool for converting FLEx datasets to Xigt. Woodley Packard also provided a great deal of assistance to us using FFTB to do treebanking.

We are deeply indebted to the linguists who shared their datasets with us (all listed in Appendix C) and generously answered our questions, as well as to the speaker communities they have worked with.

In addition, we are grateful for the feedback on this paper that we received from Olga Zamaraeva, Woodley Packard, Angelina McMillan-Major, Gina-Anne Levow, Edith Aldridge and our anonymous reviewers.

This material is based upon work supported by the National Science Foundation under Grant No. BCS-1561833 (PI Bender).

References

Ackema, Peter, Patrick Brandt, Maaike Schoorlemmer, and Fred Weerman, editors. 2006. *Arguments and Agreement*. Oxford University Press, Oxford.

Acri, Andrea. 2018. Draft of an in-progress critical edition of chapter 3 of the Bhuvanakośa prepared for the 4th International Intensive Course in Old Javanese,

Yogyakarta. 15–29 July 2018 (used with the author's permission).

Agić, Željko, Anders Johannsen, Barbara Plank, Héctor Martínez Alonso, Natalie Schluter, and Anders Søgaard. 2016. Multilingual projection for parsing truly low-resource languages. *Transactions of the Association for Computational Linguistics*, 4:301–312.

de Almeida, Tifa, Youyun Zhang, Kristen Howell, and Emily M Bender. 2019. Feature comparison across typological resources. *Unpublished abstract, presented at TypNLP*.

Bender, Emily M. 2008a. Evaluating a crosslinguistic grammar resource: A case study of Wambaya. In *Proceedings of ACL-08: HLT*, pages 977–985, Columbus. Association for Computational Linguistics.

Bender, Emily M. 2008b. Grammar engineering for linguistic hypothesis testing. In *Proceedings of the Texas Linguistics Society X Conference: Computational Linguistics for Less-Studied Languages*, pages 16–36, Stanford. CSLI Publications.

Bender, Emily M. 2008c. Radical non-configurationality without shuffle operators: An analysis of wambaya. In *Proceedings of the International Conference on Head-Driven Phrase Structure Grammar*, pages 6–24, Stanford. CSLI Publications.

Bender, Emily M. 2010. Reweaving a grammar for Wambaya: A case study in grammar engineering for linguistic hypothesis testing. *Linguistic Issues in Language Technology*, 3(3):1–34.

Bender, Emily M, Joshua Crowgey, Michael Wayne Goodman, and Fei Xia. 2014. Learning grammar specifications from IGT: A case study of Chintang. In *Proceedings of the 2014 Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 43–53, Baltimore. Association for Computational Linguistics.

Bender, Emily M, Scott Drellishak, Antske Fokkens, Laurie Poulson, and Safiyyah Saleem. 2010. Grammar customization. *Research on Language & Computation*, 8(1):23–72. 10.1007/s11168-010-9070-1.

Bender, Emily M, Dan Flickinger, and Stephan Oepen. 2002. The grammar matrix: An open-source starter-kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars. In *Proceedings of the Workshop on Grammar Engineering and Evaluation at the 19th International Conference on Computational Linguistics*, pages 8–14, Taipei.

Bender, Emily M, Dan Flickinger, Stephan Oepen, Woodley Packard, and Ann Copestake. 2015. Layers of interpretation: On grammar and compositionality. In *Proceedings of the 11th International Conference on Computational Semantics*, pages 239–249, London. Association for Computational Linguistics.

Bender, Emily M, Sumukh Ghodke, Timothy Baldwin, and Rebecca Dridan. 2012. From database to treebank: Enhancing hypertext grammars with grammar engineering and treebank search. In Sebastian Nordhoff and Karl-Ludwig G. Poggeman, editors, *Electronic Grammaticography*, pages 179–206. University of Hawai'i Press, Honolulu.

Bender, Emily M and Jeff Good. 2005. Implementation for discovery: A bipartite lexicon to support morphological and syntactic analysis. In *Proceedings from the Panels of the Forty-First Meeting of the Chicago Linguistic Society*, pages 1–15.

Bender, Emily M, Michael Wayne Goodman, Joshua Crowgey, and Fei Xia. 2013. Towards creating precision grammars from interlinear glossed text: Inferring large-scale typological properties. In *Proceedings of the 7th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 74–83, Sofia. Association for Computational Linguistics.

Bickel, Balthasar, Bernard Comrie, and Martin Haspelmath. 2008. The Leipzig glossing rules: Conventions for interlinear morpheme-by-morpheme glosses. Max Planck Institute for Evolutionary Anthropology and Department of Linguistics, University of Leipzig.

Bickel, Balthasar, Martin Gaenszle, Novel Kishore Rai, Vishnu Singh Rai, Elena Lieven, Sabine Stoll, G. Banjade, T. N. Bhatta, N Paudyal, J Pettigrew, and M Rai, I. P.and Rai. 2013a. Durga. https://corpus1.mpi.nl/qfs1/media-archive/dobes_data/ChintangPuma/Chintang/Narratives/Annotations/durga_exp.tbt Accessed: 2013.

Bickel, Balthasar, Sabine Stoll Stoll, Martin Gaenszle, Novel Kishor Rai, Elena Lieven, Goma Banjade, Toya Nath Bhatta, Netra Prasad Paudyal, Judith Pettigrew, Ichchha Purna Rai, Manoj Rai, Taras Zakharko, and Robert Schikowski. 2013b. Audiovisual corpus of the chintang language, including a longitudinal corpus of language acquisition by six children, paradigm sets, grammar sketches, ethnographic descriptions, and photographs.

Bierwisch, Manfred. 1963. *Grammatik des deutschen Verbs*, volume II of *Studia Grammatica*. Akademie Verlag.

Bird, Steven. 2022. Local languages, third spaces, and other high-resource scenarios. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7817–7829, Dublin, Ireland. Association for Computational Linguistics.

Bod, Rens. 2009. From exemplar to grammar: A probabilistic analogy-based model of language learning. *Cognitive Science*, 33(5):752–793.

Booij, Geert Evert. 2002. *The morphology of Dutch*. Oxford University Press on Demand.

Bouma, Gosse, JM van Koppen, Frank Landsbergen, JEJM Odijk, Ton van der Wouden, and Matje van de Camp. 2015. Enriching a descriptive grammar with treebank queries. In *Proceedings of the Fourteenth International Workshop on Treebanks and Linguistic Theories (TLT14)*, volume 14, pages 13–25.

Bowern, Claire. 2011. Sivisa Titan: sketch grammar, texts, vocabulary based on material collected by P. Josef Meier and Po Minis. *Oceanic Linguistics Special Publications*, 38:iii–466.

Bowern, Claire. 2012. *A grammar of Bardi*, volume 57. Walter de Gruyter.

Bowern, Claire. 2019. Titan materials. *Digital collection managed by PARADISEC [Open Access]*. (Accessed January 2019).

Buys, Jan and Phil Blunsom. 2017. Robust incremental neural semantic graph parsing. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1215–1226, Vancouver, Canada. Association for Computational Linguistics.

Carter, David. 1997. The TreeBanker: a tool for supervised training of parsed corpora. In *Computational Environments for Grammar Development and Linguistic Engineering*.

Chelliah, Shobhana Lakshmi. 2011. *A grammar of Meithei*, volume 17. Walter de Gruyter.

Chelliah, Shobhana Lakshmi. 2019. Meithei texts. Manipur Digital Resources in UNT Digital Library. University of North Texas Libraries. (Accessed August 2019).

Chen, Yufei, Weiwei Sun, and Xiaojun Wan. 2018. Accurate SHRG-based semantic parsing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 408–418, Melbourne, Australia. Association for Computational Linguistics.

Chomsky, Noam. 1995. *The Minimalist Program*. MIT Press, Cambridge.

Clark, Stephen and James R. Curran. 2004. Parsing the WSJ using CCG and log-linear models. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 103–110, Barcelona, Spain.

Comrie, Bernard. 1989. *Language Universals & Linguistic Typology*, second edition. University of Chicago, Chicago.

Conrad, Elizabeth. 2021. Tracing and reducing lexical ambiguity in automatically inferred grammars. Master's thesis, University of Washington.

Copestake, Ann. 2002a. Definitions of typed feature structures. In Stephan Oepen, Dan Flickinger, Junichi Tsujii, and Hans Uszkoreit, editors, *Collaborative Language Engineering*, pages 227–230. CSLI Publications, Stanford.

Copestake, Ann. 2002b. *Implementing typed feature structure grammars*. CSLI publications Stanford.

Copestake, Ann, Dan Flickinger, Carl Pollard, and Ivan A Sag. 2005. Minimal Recursion Semantics: An introduction. *Research on Language and Computation*, 3(2-3):281–332.

Corbett, Greville G. 1991. Gender. *Cambridge: CUP*.

Corbett, Greville G. 2000. Number. *Cambridge: CUP*.

da Costa, Luis Morgado, Francis Bond, and Xiaoling He. 2016. Syntactic well-formedness diagnosis and error-based coaching in computer assisted language learning using machine translation. In *Proceedings of the 3rd Workshop on Natural Language Processing Techniques for Educational Applications (NLPTEA2016)*, pages 107–116.

Cowell, Andrew. 2018. Arapaho text database. Version 1, 2018. University of Colorado, Department of Linguistics (Accessed at https://github.com/Adamits/arapaho_library/tree/master/data February 2020).

Cowell, Andrew and Alonzo Moss Sr. 2011. *The Arapaho language*. University Press of Colorado.

Crowgey, Joshua. 2019. *Braiding Language (by Computer): Lushootseed Grammar Engineering*. Ph.D. thesis, University of Washington.

Crowgey, Joshua David. 2012. The syntactic exponence of sentential negation: A model for the LinGO Grammar Matrix. Master's thesis, University of Washington.

Crysmann, Berthold and Woodley Packard. 2012. Towards efficient HPSG generation for German, a non-configurational language. In *Proceedings of the 24th International Conference on Computational Linguistics*, pages 695–710.

Cysouw, Michael. 2013. Inclusive/exclusive distinction in independent pronouns. In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Östen Dahl. 1979. Typology of sentence negation. *Linguistics*, 17(1-2):79–106.

Dedrick, John M and Eugene H Casad. 1999. *Sonora Yaqui Language Structures*. University of Arizona Press.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Dixon, RMW. 1994. *Ergativity*. Cambridge University Press, Cambridge.

Donet, Charles. 2014a. The importance of verb salience in the followability of Lezgi oral narratives. Master's thesis, Dallas International University.

Donet, Charles. 2014b. Lezgi oral narratives. Dallas International University. Unpublished FieldWorks (FLEx) project. (Accessed August 2019).

Drellishak, Scott. 2004. A survey of coordination strategies in the world's languages. Master's thesis, University of Washington.

Drellishak, Scott. 2009. *Widespread but not universal: Improving the typological coverage of the Grammar Matrix*. Ph.D. thesis, University of Washington.

Drellishak, Scott and Emily M Bender. 2005. A coordination module for a crosslinguistic grammar resource. In *The Proceedings of the 12th International Conference on Head-Driven Phrase Structure Grammar, Department of Informatics, University of Lisbon*, pages 108–128, Stanford. CSLI Publications.

Dridan, Rebecca and Stephan Oepen. 2011. Parser evaluation using elementary dependency matching. In *Proceedings of the 12th International Conference on Parsing Technologies*, pages 225–230.

Dryer, Matthew S. 2005. Negative morphemes. In Martin Haspelmath, Matthew S. Dryer, David Gil, and Bernard Comrie, editors, *The World Atlas of Linguistic Structures (WALS)*, pages 454–457. Oxford University Press, Oxford.

Dryer, Matthew S. 2013a. Expression of pronominal subjects. In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig. Available at https://wals.info/chapter/101, Accessed 2022-05-04.

Dryer, Matthew S. 2013b. Negative morphemes. In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig. Available at https://wals.info/chapter/112, Accessed 2022-05-04.

Dryer, Matthew S. 2013c. Order of subject, object and verb. In Matthew S. Dryer and Martin Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig. Available at https://wals.info/chapter/81, Accessed 2022-05-04.

Dryer, Matthew S. and Martin Haspelmath, editors. 2013. *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig. Available at https://wals.info/, Accessed 2022-05-04.

Eberhard, David M., Gary F. Simons, and Charles D. Fennig (eds.). 2019. Ethnologue: Languages of the World. Twenty-second edition. Available at http://www.ethnologue.com, Accessed 2022-05-04.

Flickinger, Dan. 2000. On building a more efficient grammar by exploiting types. *Natural Language Engineering*, 6 (1) (Special Issue on Efficient Processing with HPSG):15 – 28.

Flickinger, Dan. 2011. Accuracy v. robustness in grammar engineering. In Emily M Bender and Jennifer E Arnold, editors, *Language from a Cognitive Perspective: Grammar, Usage and Processing*, pages 31–50. CSLI Publications, Stanford.

Flickinger, Dan, Emily M Bender, and Stephan Oepen. 2014a. ERG semantic documentation. Accessed on 2022-05-16.

Flickinger, Dan, Emily M Bender, and Stephan Oepen. 2014b. Towards an encyclopedia of compositional semantics: Documenting the interface of the English resource grammar. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 875–881, Reykjavik. European Language Resources Association (ELRA).

Flickinger, Dan, Stephan Oepen, and Emily M Bender. 2017. Sustainable development and refinement of complex linguistic annotations at scale. In Nancy Ide and James Pustejovsky, editors, *Handbook of Linguistic Annotation*, pages 353–377. Springer Netherlands, Dordrecht.

Fokkens, Antske. 2014. *Enhancing Empirical Research for Linguistically Motivated Precision Grammars*. Ph.D. thesis, Department of Computational Linguistics, Universität des Saarlandes.

Georgi, Ryan. 2016. *From Aari to Zulu: Massively Multilingual Creation of Language Tools Using Interlinear Glossed Text*. Ph.D. thesis, University of Washington.

GOLD. 2010. General Ontology for Linguistic Description (GOLD). Bloomington, IN: Department of Linguistics (The LINGUIST List), Indiana University. Available at http://linguistics-ontology.org/, Accessed 2022-05-06.

Goodman, Michael Wayne. 2013. Generation of machine-readable morphological rules from human readable input. *Seattle: University of Washington Working Papers in Linguistics*, 30.

Goodman, Michael Wayne, Joshua Crowgey, Fei Xia, and Emily M Bender. 2015. Xigt: Extensible interlinear gloss text for natural language processing. *Language Resources and Evaluation*, 49 (2):455–485.

Han, Wenjuan, Ge Wang, Yong Jiang, and Kewei Tu. 2019. Multilingual grammar induction with continuous language identification. 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 5732–5737.

Harley, Heidi. 2019. Haiki text corpus. University of Arizona. Unpublished FieldWorks (FLEx) project. (Accessed August 2019).

Haspelmath, Martin. 2007. Coordination. In Timothy Shopen, editor, *Language typology and syntactic description*, volume 2. Cambridge University Press, Cambridge.

Hauk, Bryn. 2016–2019. Tsova-tush lexicon and texts. University of Hawai'i at Mānoa. Unpublished FieldWorks (FLEx) project. V2019.08.20. 2016–2019 (collection date).

Hauk, Bryn. 2020. *Deixis and reference tracking in Tsova-Tush*. Ph.D. thesis, University of Hawai'i at Mānoa.

Hauk, Bryn and Alice C. Harris. forthcoming. Batsbi. In Yuri Koryakov, Yury Lander, and Timur Maisak, editors, *The Caucasian languages: An international handbook*. De Gruyter Mouton.

Hellan, Lars. 2010. From descriptive annotation to grammar specification. In *Proceedings of the Fourth Linguistic Annotation Workshop*, pages 172–176, Uppsala. Association for Computational Linguistics.

Hellan, Lars and Dorothee Beermann. 2011. Inducing grammars from IGT. In *Human Language Technology Challenges for Computer Science and Linguistics.*, volume 8287 of *LTC 2011. Lecture Notes in Computer Science*. Springer.

Hellan, Lars, Tore Bruland, Elias Aamot, and Mads H Sandøy. 2013. A grammar sparrer for Norwegian. In *Proceedings of the 19th Nordic Conference of Computational Linguistics (NODALIDA 2013); May 22-24; 2013; Oslo University; Norway. NEALT Proceedings Series 16*, 085, pages 435–439. Linköping University Electronic Press.

Hewitt, John and Christopher D Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4129–4138, Minneapolis. Association for Computational Linguistics.

Hinds, John. 1986. *Japanese: Descriptive Grammar*. Routledge, New York.

Hockenmaier, Julia and Mark Steedman. 2002. Generative models for statistical parsing with Combinatory Categorial Grammar. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 335–342, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Hockenmaier, Julia and Mark Steedman. 2007. CCGbank: A corpus of CCG derivations and dependency structures extracted from the Penn Treebank. *Computational Linguistics*, 33(3):355–396.

Holton, David, Peter Mackridge, Irene Philippaki-Warburton, and Vassilios Spyropoulos. 2012. *Greek: A comprehensive grammar of the modern language*, 2nd edition. Routledge, London.

Hopper, Paul J. 1982. *Tense-aspect: Etween Semantics & Pragmatics: Containing the Contributions to a Symposium on Tense and Aspect, held at UCLA, May 1979*, volume 1. John Benjamins Publishing, Amsterdam/Philadelphia.

Howell, Kristen. 2020. *Inferring Grammars from Interlinear Glossed Text: Extracting Typological and Lexical Properties for the Automatic Generation of HPSG Grammars*. Ph.D. thesis, University of Washington.

Howell, Kristen, Emily M Bender, Michel Lockwood, Fei Xia, and Olga Zamaraeva. 2017. Inferring case systems from IGT: Impacts and detection of variable glossing practices. pages 67–75.

Howell, Kristen and Olga Zamaraeva. 2018. Clausal modifiers in the Grammar Matrix. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2939–2952.

Howell, Kristen, Olga Zamaraeva, and Emily M Bender. 2018. Nominalized clauses in the Grammar Matrix. In *Proceedings of the 25th International Conference on Head-Driven Phrase Structure Grammar, University of Tokyo*.

Indurkhya, Sagar. 2020. Inferring Minimalist grammars with an SMT-solver. In *Proceedings of the Society for Computation in Linguistics*, volume 3.

Inman, David. 2015. Pronoun incorporation in Matsigenka. Unpublished Manuscript, available at http://compling.hss.ntu.edu.sg/events/2015-hpsg/pdf/Inman.pdf, Accessed 2022-05-06.

Inman, David. 2019a. *Multi-predicate Constructions in Nuuchahnulth*. Ph.D. thesis, University of Washington.

Inman, David. 2019b. Nuuchahnulth texts. University of Washington. Unpublished FieldWorks (FLEx) project. (Accessed March 2019).

Kaplan, Ronald M and Joan Bresnan. 1982. Lexical-functional grammar: A formal system for grammatical representation. In Joan Bresnan, editor, *The Mental Representation of Grammatical Relations (MIT Press Series on Cognitive Theory and Mental Representation)*, pages 173–281. The MIT Press, Cambridge.

Kate, Rohit J and Raymond J Mooney. 2006. Using string-kernels for learning semantic parsers. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 913–920. Association for Computational Linguistics.

Kate, Rohit J, Yuk Wah Wong, and Raymond J Mooney. 2005. Learning to transform natural to formal languages. In *Proceedings of the 1st AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, pages 1062–1068.

Kaufman, Daniel, Husniya Khujamyorova, and Ross Perlin. 2020. Wakhi texts. *Digital collection managed by KRATYLOS*. Uploaded from www.elalliance.org, Wakhi. In Finkel, R. and Kaufman, D., Kratylos: Unified Linguistic Corpora from Diverse Data Sources. Uploaded April 28, 2020 and retrieved from https://www.cs.uky.edu/ raphael/ela/ on May 20 2020.

Kenesei, István, Robert M Vago, and Anna Fenyvesi. 2002. *Hungarian*, 1st edition. Routledge, London.

Klein, Dan and Christopher D Manning. 2001. Natural language grammar induction using a constituent-context model. In *Advances in neural information processing systems 14*, pages 35–42.

Klein, Dan and Christopher D Manning. 2002. A generative constituent-context model for improved grammar induction. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 128–135, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Klein, Dan and Christopher D Manning. 2004. Corpus-based induction of syntactic structure: Models of dependency and constituency. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 478–485, Barcelona, Spain.

Kornfilt, Jaklin. 1997. *Turkish*. Routledge, London.

Kratochvíl, František. 2007. *A grammar of Abui*. LOT, Utrecht.

Kratochvíl, František. 2019. Abui Corpus. Electronic Database: Unpublished toolbox project (accessed March 2019). Nanyang Technological University, Singapore.

Krotov, Alexander, Robert Gaizauskas, and Yorick Wilks. 1994. Acquiring a stochastic context-free grammar from the Penn Treebank. In *Proceedings of the Irish Conference on NLP, Dublin*.

Krotov, Alexander, Mark Hepple, Robert Gaizauskas, and Yorick Wilks. 1998. Compacting the Penn Treebank Grammar. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (ACL-1998)*, pages 699–703, Montreal.

Kwiatkowski, Tom, Luke Zettlemoyer, Sharon Goldwater, and Mark Steedman. 2011. Lexical generalization in CCG grammar induction for semantic parsing. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1512–1523. Association for Computational Linguistics.

Letcher, Ned and Timothy Baldwin. 2013. Constructing a phenomenal corpus: Towards detecting linguistic phenomena in precision grammars. In *Proceedings of the Workshop on High-level Methodologies for Grammar Engineering at ESSLLI 2013*, pages 25–36.

Li, Charles N and Sandra A Thompson. 1989. *Mandarin Chinese: A functional reference grammar*. University of California Press, Berkeley/Los Angeles.

Lockwood, Michael. 2016. Automated gloss mapping for inferring grammatical properties. Master's thesis, University of Washington.

Marcus, Mitchell, 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*, pages 114–119. Association for Computational Linguistics.

Master, Alfred. 1946. The zero negative in Dravidian. *Transactions of the Philological Society*, 45(1):137–155.

Meira, Sérgio. 2020. Hixkaryana lexicon and texts. Unpublished Toolbox project. (Accessed March 2020).

Michael, Lev, Christine Beier, Zachary O'Hagan, (compilers), Haroldo Vargas, José Vargas, and (authors). 2013. Matsigenka text corpus (version june 2013; FLEx database and LaTeX interlinear output).

Michael, Lev David. 2008. *Nanti evidential practice: Language, knowledge, and social action in an Amazonian society*. Ph.D. thesis, University of Texas Austin.

Miestamo, Matti. 2008. *Standard negation: The negation of declarative verbal main clauses in a typological perspective*, volume 31. Walter de Gruyter, Berlin.

Miyaoka, Osahito. 2012. *A Grammar of Central Alaskan Yupik (CAY)*, volume 58. Walter de Gruyter, Berlin.

Monachesi, Paola. 1996. *A grammar of Italian clitics*. ITK Dissertations Series 1996-1.

Müller, Stefan. 1999. *Deutsche Syntax deklarativ. Head-Driven Phrase Structure Grammar für das Deutsche (Linguistische Arbeiten 394)*. Max Niemeyer, Tübingen.

Müller, Stefan. 2015. The CoreGram project: Theoretical linguistics, theory development and verification. *Journal of Language Modelling*, 3(1):21–86.

Müller, Stefan, Anne Abeillé, Robert D. Borsley, and Jean-Pierre Koenig, editors. 2021. *Head-Driven Phrase Structure Grammar: The handbook (Empirically Oriented Theoretical Morphology and Syntax 9)*. Language Science Press, Berlin. https://doi.org/10.5281/zenodo.5543318.

Muysken, Pieter, Harald Hammarström, Olga Krasnoukhova, Neele Müller, Joshua Birchall, Simon van de Kerke, Loretta O'Connor, Swintha Danielsen, Rik van Gijn, and George Saad, editors. 2016. *South American Indigenous Language Structures (SAILS) Online*. Max Planck Institute for the Science of Human History. Available at https://sails.clld.org, Accessed 2022-05-04.

Newman, Paul. 2000. *The Hausa language: An encyclopedic reference grammar*. Yale University Press, New Haven.

Nielsen, Elizabeth and Emily M Bender. 2018. Modeling adnominal possession in multilingual grammar engineering. In *Proceedings of the 25th International Conference on Head-Driven Phrase Structure Grammar, University of Tokyo*, pages 140–153, Stanford. CSLI Publications.

Nielsen, Elizabeth K. 2018. Modeling adnominal possession in the LinGO Grammar Matrix. Master's thesis, University of Washington.

Nivre, Joakim, Marie-Catherine De Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. 2016. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1659–1666.

Noji, Hiroshi, Yusuke Miyao, and Mark Johnson. 2016. Using left-corner parsing to encode universal structural constraints in grammar induction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 33–43.

Nordlinger, Rachel. 1998. *A Grammar of Wambaya, Northern Australia*. Pacific Linguistics, Canberra.

Oepen, Stephan. 2001. [incr tsdb()] — Competence and performance laboratory. User manual. Technical report, Computational Linguistics — Saarland University, Saarbrücken.

Oepen, Stephan, Kristina Toutanova, Stuart Shieber, Chris Manning, Dan Flickinger, and Thorsten Brants. 2002. The LinGO Redwoods treebank. Motivation and preliminary applications. In *Proceedings of the 19th International Conference on Computational Linguistics*, Taipei.

O'Hagan, Zachary. 2018. The syntax of Matsigenka object-marking. *Berkeley Papers in Formal Linguistics*, 1(1).

Packard, Woodley. 2015. Full Forest Treebanking. Master's thesis, University of Washington.

Pollard, Carl and Ivan A Sag. 1994. *Head-Driven Phrase Structure Grammar (Studies in Contemporary Linguistics)*. University of Chicago Press, Chicago.

Poulson, Laurie. 2011. Meta-modeling of tense and aspect in a cross-linguistic grammar engineering platform. *University of Washington Working Papers in Linguistics (UWWPL)*, 28.

Pustejovsky, James, José M Castaño, Robert Ingria, Roser Saurí, Robert J Gaizauskas, Andrea Setzer, and Graham Katz. 2003. TimeML: Robust specification of event and temporal expressions in text. In *Proceedings of the IWCS-5 Fifth International Workshop on Computational Semantics*.

Rogers, Chris. 2010. Fieldworks language explorer (FLEx) 3.0. *Language Documentation & Conservation*, 4:78–84.

Saleem, Safiyyah. 2010. Argument optionality: A new library for the Grammar Matrix customization system. Master's thesis, University of Washington.

Saleem, Safiyyah and Emily M Bender. 2010. Argument optionality in the LinGO Grammar Matrix. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 1068–1076. Association for Computational Linguistics.

Sanchez, Jose, Alex Trueman, Maria Florez Leyva, Santos Leyva Alvarez, Mercedes Tubino Blanco, Hyun-Kyoung Jung, Louise St. Amour, and Heidi Harley. 2015. *An Introduction to Hiaki Grammar*. University of Arizona Press, Tucson.

Sarveswaran, Kengatharaiyer, Gihan Dias, and Miriam Butt. 2019. Using meta-morph rules to develop morphological analysers: A case study concerning Tamil. In *Proceedings of the 14th International Conference on Finite-State Methods and Natural Language Processing*, pages 76–86, Dresden. Association for Computational Linguistics.

Schikowski, Robert. 2013. *Object-conditioned differential marking in Chintang and Nepali*. Ph.D. thesis, University of Zurich.

Schrock, Terrill B. 2014. *A Grammar of Ik (Icé-tód): Northeast Uganda's Last Thriving Kuliak Language*. LOT, Utrecht.

Shi, Haoyue, Jiayuan Mao, Kevin Gimpel, and Karen Livescu. 2019. Visually grounded neural syntax acquisition. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1842–1861, Florence. Association for Computational Linguistics.

Siegel, Melanie, Emily M Bender, and Francis Bond. 2016. *Jacy: An implemented grammar of Japanese*. CSLI Publications, Stanford.

Siewierska. 2004. *Person*. Cambridge University Press, Cambridge.

SIL International. 2015. Field Linguist's Toolbox. Lexicon and corpus management system with a parser and concordancer; Available at https://software.sil.org/fieldworks/download/, Accessed 2022-05-04.

Simov, Kiril. 2002. Grammar extraction and refinement from an HPSG corpus. In *Proceedings of the ESSLLI Workshop on Machine Learning Approaches in Computational Linguistics*, pages 38–55.

Smith, Noah A and Jason Eisner. 2006. Annealing structural bias in multilingual weighted grammar induction. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (ACL/COLING 2006)*, pages 569–576, Sydney. Association for Computational Linguistics.

Sneddon, James Neil, K Alexander Adelaar, Dwi N Djenar, and Michael Ewing. 2012. *Indonesian: A comprehensive grammar*. Routledge, Oxfordshire.

Sohn, Ho-Min. 1994. *Korean: A Descriptive Grammar*. Routledge, London/New York.

Stabler, Edward. 1996. Derivational minimalism. In *International Conference on Logical Aspects of Computational Linguistics*, pages 68–95, Berlin/Heidelberg. Springer.

Sulkala, Helena and Merja Karjalainen. 1992. *Finnish*. Routledge, London/New York.

Sylak-Glassman, John, Christo Kirov, David Yarowsky, and Roger Que. 2015. A language-independent feature schema for inflectional morphology. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 674–680.

Thieberger, Nick. 2006a. Dictionary and texts in South Efate. *Digital collection managed by PARADISEC [Open Access]*. (Accessed March 2019).

Thieberger, Nick. 2006b. *A grammar of South Efate: an Oceanic language of Vanuatu*, volume 33. University of Hawai'i Press, Honolulu.

de Urbina, Jon Ortiz. 1989. *Parameters in the grammar of Basque: A GB approach to Basque syntax*. Foris, Dordrecht/Providence.

Wax, David. 2014. Automated grammar engineering for verbal morphology. Master's thesis, University of Washington.

Xia, Fei. 1999. Extracting tree adjoining grammars from bracketed corpora. In *Proceedings of 5th Natural Language Processing Pacific Rim Symposium (NLPRS-1999)*, Beijing.

Xia, Fei and William D. Lewis. 2007. Multilingual structural projection across interlinear text. In *Proceedings of the Conference on Human Language Technologies (HLT/NAACL 2007)*, pages 452–459, Rochester.

Xia, Fei, William D. Lewis, Michael Wayne Goodman, Glenn Slayden, Ryan Georgi, Joshua Crowgey, and Emily M Bender. 2016. Enriching a massively multilingual database of interlinear glossed text. *Language Resources and Evaluation*, 50:321–349.

Zamaraeva, Olga. 2016. Inferring morphotactics from interlinear glossed text: combining clustering and precision grammars. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 141–150.

Zamaraeva, Olga. 2021. *Assembling Syntax: Modeling Constituent Questions in a Grammar Engineering Framework*. Ph.D. thesis, University of Washington.

Zamaraeva, Olga, Kristen Howell, and Emily M Bender. 2019a. Handling cross-cutting properties in automatic inference of lexical classes: A case study of Chintang. In *Proceedings of the 3rd Workshop on the Use of Computational Methods in the Study of Endangered Languages*, volume 1 Papers, pages 28–38, Honolulu, Hawai'i.

Zamaraeva, Olga, Kristen Howell, and Emily M Bender. 2019b. Modeling clausal complementation for a grammar engineering resource. In *Proceedings of the Society for Computation in Linguistics*, volume 2, page Article 6.

Zamaraeva, Olga, František Kratochvíl, Emily M Bender, Fei Xia, and Kristen Howell. 2017. Computational support for finding word classes: A case study of Abui. In *Proceedings of the 2nd Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 130–140.

Zamaraeva, Olga, TJ Trimble, Kristen Howell, Michael Wayne Goodman, Antske Fokkens, Guy Emerson, Chris Curtis, and Emily M Bender. forthcoming. 20 years of the Grammar Matrix: Cross-linguistic hypothesis testing of increasingly complex interactions. *Journal of Language Modeling*.

Zanzotto, Fabio Massimo, Andrea Santilli, Leonardo Ranaldi, Dario Onorati, Pierfrancesco Tommasino, and Francesca Fallucchi. 2020. KERMIT: Complementing transformer architectures with encoders of explicit syntactic interpretations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 256–267.

Zhang, Songyang, Linfeng Song, Lifeng Jin, Kun Xu, Dong Yu, and Jiebo Luo. 2021. Video-aided unsupervised grammar induction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1513–1524, Online. Association for Computational Linguistics.

Zhao, Yanpeng and Ivan Titov. 2020. Visually grounded compound PCFGs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4369–4379, Online. Association for Computational Linguistics.

Zwicky, Arnold, Joyce Friedman, Barbara C. Hall, and D.E. Walker. 1965. The MITRE syntactic analysis procedure for transformational grammars. In *Proceedings Fall Joint Computer Conference*, volume 67, Pt 1, pages 317–326.

Zwicky, Arnold M and Geoffrey K Pullum. 1983. Cliticization vs. inflection: English n't. *Language*, 59(3):502–513.

Zymla, Mark-Matthias. 2017. Comprehensive annotation of cross-linguistic variation in tense and aspect categories. In *IWCS 2017-12th International Conference on Computational Semantics-Long papers*.

A Data Repositories

Alaskan Native Languages Archive (ANLA)
https://www.uaf.edu/anla/
Archive of Indigenous Languages in Latin America (AILLA)
http://www.ailla.utexas.org/site/welcome.html
Endangered Languages Archive (ELAR)
http://elar.soas.ac.uk/
Kaipuleohone
https://scholarspace.manoa.hawaii.edu/handle/10125/4250
Kratylos
https://www.kratylos.org/~kratylos/home.cgi
Multi-CAST
https://multicast.aspra.uni-bamberg.de/
ODIN
http://depts.washington.edu/uwcl/odin/
Pacific and Regional Archive for Digital Sources (PARADISEC)
http://www.paradisec.org.au/

B Code and Project Repositories

ACE
http://sweaglesw.org/linguistics/ace/
AGGREGATION, BASIL
https://git.ling.washington.edu/agg
DELPH-IN
www.delph-in.net
INTENT
https://github.com/rgeorgi/INTENT2
FFTB
http://moin.delph-in.net/FftbTop
Grammar Matrix
http://matrix.ling.washington.edu/index.html
MOM
https://git.ling.washington.edu/agg/mom
Xigt
https://github.com/xigt/xigt

C Languages, Corpora and Descriptive Resources

The languages and corpora used for this research are listed in the table below, together with any descriptive resources we consulted during BASIL's development and evaluation.

	Language	iso	Corpus	Descriptive Resource
	Development			
1	Abui	abz	Kratochvíl 2019	Kratochvíl 2007
2	Chintang	ctn	Bickel et al. 2013b	Schikowski 2013
3	Matsigenka	mcb	Michael et al. 2013	Michael 2008
4	Nuuchahnulth	nuk	Inman 2019b	Inman 2019a
5	Wambaya	wmb	Nordlinger 1998	Nordlinger 1998
6	Haiki	yaq	Harley 2019	Sanchez et al. 2015
				Dedrick and Casad 1999
7	Lezgi	lez	Donet 2014b	Donet 2014a
8	Meithei	mni	Chelliah 2019	Chelliah 2011
9	Tsova-Tush	bbl	Hauk 2016–2019	Hauk and Harris forthcoming
				Hauk 2020
	Consulted			
10	Bardi	bcj	Bowern 2012	Bowern 2012
11	Ik	ikx	Schrock 2014	Schrock 2014
12	Old Javanese	jav	Acri 2018	
13	Yup'ik	esu	Miyaoka 2012	Miyaoka 2012
14	Basque	eus	Xia et al. 2016	de Urbina 1989
15	Dutch	nld	Xia et al. 2016	Booij 2002
16	Finnish	fin	Xia et al. 2016	Sulkala and Karjalainen 1992
17	Greek	ell	Xia et al. 2016	Holton et al. 2012
18	Hausa	hau	Xia et al. 2016	Newman 2000
19	Hungarian	hun	Xia et al. 2016	Kenesei et al. 2002
20	Indonesian	ind	Xia et al. 2016	Sneddon et al. 2012
21	Italian	ita	Xia et al. 2016	Monachesi 1996
22	Japanese	jpn	Siegel et al. 2016	Siegel et al. 2016
			Xia et al. 2016	Hinds 1986
23	Korean	kor	Xia et al. 2016	Sohn 1994
24	Mandarin	cmn	Xia et al. 2016	Li and Thompson 1989
25	Polish	pol	Xia et al. 2016	
26	Russian	rus	Xia et al. 2016	
27	Turkish	tur	Xia et al. 2016	Kornfilt 1997
	Held Out			
28	Arapaho	arp	Cowell 2018	Cowell and Moss Sr 2011
29	Hixkaryana	hix	Meira 2020	
30	South Efate	erk	Thieberger 2006a	Thieberger 2006b
31	Titan	ttv	Bowern 2019	Bowern 2011
32	Wakhi	wbl	Kaufman et al. 2020	

Bias Identification and Attribution in NLP Models With Regression and Effect Sizes

Erenay Dayanik, IMS, University of Stuttgart, Germany erenay.dayanik@ims.uni-stuttgart.de
Ngoc Thang Vu, IMS, University of Stuttgart, Germany ngoc-thang.vu@ims.uni-stuttgart.de
Sebastian Padó, IMS, University of Stuttgart, Germany sebastian.pado@ims.uni-stuttgart.de

Abstract There is a growing awareness that many NLP systems incorporate biases of various types (e.g., regarding gender or race) which can cause significant social harm. At the same time, the techniques often used for the statistical analysis of biases in NLP systems are still relatively basic. Typically, studies test for the presence of a significant difference between two levels of a single bias variable (e.g., gender: male vs. female) without attention to potential confounders, and do not quantify the importance of the bias variable. This article proposes to analyze bias in the output of NLP systems using multivariate regression models. Such models provide a robust and more informative alternative which (a) generalizes to multiple bias variables, (b) can take covariates into account, (c) can be combined with measures of effect size to quantify the size of bias. Jointly, these effects contribute to a statistically more robust identification and attribution of bias that can be used to diagnose system behavior and extract informative examples. We demonstrate the benefits of our method by analyzing a range of current NLP models on two tasks, namely one regression task (emotion intensity prediction) and one classification task (coreference resolution).

1 Introduction

Machine learning has been a major driver of innovation in natural language processing since the 1990s, but only the last decade has seen the widespread deployment of NLP methods for use by non-experts: Applications such as neural machine translation (Wu et al., 2016) or voice assistants (Këpuska and Bohouta, 2018) are now routinely available through end users' mobile phones, and NLP methods are increasingly used in domains outside computer science such as police work (Sun et al., 2021) and recruiting (Singh et al., 2010).

Such systems are, from a user perspective, black boxes whose predictions are generally taken at face value. This makes the question pertinent to what extent the machine learning methods underlying these NLP models are *fair*, or, on the contrary, to what extent they are subject to *biases* which impact their predictions. More formally, Friedman and Nissenbaum (1996) defined biased computer systems as systems that "*systematically and unfairly discriminate against certain individuals or groups of individuals in favor of others*"; (see Mehrabi et al. (2021) for a very similar definition). Clearly, such biases have the potential to cause concrete *harm* for the disadvantaged groups or individuals (Bender and Friedman, 2018; Blodgett et al., 2020) and must be observed and controlled as far as possible.

A practical aspect of bias analysis, which the above definition leaves open, is whether discrimination is measured "in vitro" (at the level of system performance) or "in vivo" (at the level of real world consequences). In line with the majority of NLP studies on bias, the present study focusses on bias measured "in vitro", i.e., in the form of systematic differences in system performance across groups. We acknowledge the need to better understand how such "in vitro" bias translates into "in vivo" real-world consequences, and argue below that the methods we propose offer a first step in this direction.

A quickly growing body of studies has indeed found that biases are, unfortunately, pervasive in NLP systems (Mehrabi et al., 2021). One of the first studies on bias, Bolukbasi et al. (2016) analyzed similarity relations in word embeddings and found a substantial *gender bias*, as a result of which, e.g., *woman* was more similar to *nurse* than *doctor*, while *man* was more similar to *doctor* than *nurse*. Davidson et al. (2019) found systematic and substantial racial biases in five Twitter datasets annotated for offensive language detection, where African American English tweets were overclassified as hateful compared with Standard American English, and Díaz et al. (2018) found a significant age bias in many sentiment

analysis algorithms, attributing less positive attitudes to older participants. See Section 2 for more details.

Consequently, dealing with biases is rapidly becoming a major high-level consideration in the design and development of NLP systems. The three main bias-related tasks are (a) bias identification (is bias present?), (b) bias attribution (where does the bias come from?) and (c) bias mitigation (how to minimize the bias?). In this article, we focus on the first two tasks, bias identification and attribution.

Following the definition given above, the identification of "in vitro" bias involves the establishment of systematic differences in system performance between two parallel stimuli sets for different levels of a *bias variable* such as gender or race. Put simply, the question is: Does, e.g., the gender of an author have a systematic influence on the output of an NLP system (e.g., are texts written by women predicted to be less positive?), or on the quality of the NLP system? (E.g., are text written by women analyzed less reliably?)

This question can be answered using statistical analysis techniques of increasing complexity, shown in Table 1. To our knowledge, all existing studies on bias fall into either the first or the second group. Studies in the first group only quantify the *performance differences*. For instance, studies investigating gender bias have generated predictions for sentence pairs which differ only in gendered expressions (e.g., cf. Table 2) and reported the difference between these sets (Zhao et al., 2018; Stanovsky et al., 2019). Without considering between-system and between-item variance, it is not clear that such differences are indeed *systematic*, as required by the definition of bias from above. For this reason, studies from the second group additionally carry out *hypothesis tests*, typically t-tests, to assess the statistical significance of the differences (Kiritchenko and Mohammad, 2018).

Although this procedure is conceptually simple and straightforward, it is problematic for two reasons. First, the pairwise hypothesis tests that are being employed in existing work assume that differences between the two sets of stimuli are due to the selected bias variable. They cannot ensure that the putative effect of bias is not due to a *covariate* that acts as a *confounding variable* (McNamee, 2005). For instance, studies on gender bias often use sets of male and female names as part of their stimulus sets (cf. Table 2). Across genders, these names may differ in the average age of the bearer, or simply in their frequency in texts, both of which may influence the performance of NLP systems (Díaz et al., 2018; Gerz et al., 2018). Similarly, author gender may be correlated with topic (Schmid, 2002; Schwemmer and Jungkunz, 2019), which can also have an impact on analyses. Therefore, even when an analysis of performance differences by gender may yield a significant performance difference, it is advisable to rule out that there are competing explanations of the difference in performance in terms of other factors.

Second, bias studies in NLP currently generally test for *statistical significance*, but very few consider *model fit* and *effect sizes* (with the notable exception of (Caliskan et al., 2017)). Significance ensures that an identified effect is not a random fluke, but does not quantify how much of the variance in the predictions is due to the bias. Given a sufficiently large dataset, even very small differences that are not practically relevant can reach significance. In contrast, the computation of effect sizes permits users to understand the practical impact of biases (Sullivan and Feinn, 2012), and is therefore arguably a first step moving from bias "in vitro" towards bias "in vivo".

In this article, we propose that these two limitations can be alleviated by adopting *multivariate regression models* such as linear and logistic regression for bias identification. This solution has already become standard in neighboring disciplines like linguistics and psychology. In regression models, bias variables and their covariates form the independent variables, and the predictions of NLP systems for corresponding instances constitute the dependent variable of the equation. As the last column in Table 1 presents, multivariate regression models have many advantages over the other two approaches for bias analysis: (a), they generalize to multiple bias variables; (b), they offer a principled treatment of covariates; (c), they come with measures of effect size that quantify the size of the bias, and (d), they provide a rich diagnosis of system behavior and can be mined easily to extract informative datapoints. In NLP, regression models of various kinds have been used widely as *predictive* models. In our paper, we focus on their use as *explanatory* models, where the focus is on building an interpretable model. Models of this type have been applied to analyze the influence of task and data properties on the performance of sequence labeling models (Papay et al., 2020) or the influence of various textual properties of author responses on the peer review process (Gao et al., 2019). We would like to stress that the goal of this procedure is not to "explain away" biases, but rather to propose a more stringent procedure to identify them, in order to strengthen their empirical standing.

Our concrete contributions are as follows:

- We identify limitations of the statistical methods that are currently applied for bias identification (Section 1).
- We propose a workflow and a set of best practices for designing, computing and interpreting multivariate regression models for this task (Section 3).
- We apply our workflow to two tasks: emotion intensity prediction, a regression task (Section 4) and coreference resolution, a classification task

	Performance Difference (Rudinger et al. 2018, Zhao et al. 2018, etc.)	Performance Difference plus Hypothesis Testing (Caliskan et al. 2017, Kiritchenko et al. 2018, etc.)	Regression Modeling with Effect Sizes (Ours)
Assessing statistical significance	-	+	+
Quantifying the impact of multiple variables	-	-	+
Diagnosing system behavior	+	+	+

Table 1: Comparison of different approaches to statistical analysis of bias.

(Section 5). Our results are in line with established findings, but permit a more nuanced and richer understanding of system behavior.

The complete code for our experiments is publicly available at `https://github.com/multireg/multireg-effect`.

2 Related Work

This section sketches the state of the art in bias analysis, More comprehensive reviews are provided by Sun et al. (2019), Blodgett et al. (2020) and Mehrabi et al. (2021).

Bias in embeddings. At the representation level, almost all state-of-the-art NLP systems use corpus-derived embeddings. These embeddings were the starting point for a lot of work on bias in NLP. Bias in embeddings is generally shown by comparing embeddings for two sets of previously established, e.g., gendered (male and female) words (e.g. *man, woman*). Bolukbasi et al. (2016) define the gender bias of a word by its projection on the difference vector between male and female embeddings; this method was found by Gonen and Goldberg (2019) to be an imperfect metric of bias. As an alternative, the WEAT benchmark (Caliskan et al., 2017) defines bias in terms of similarity to the two sets of gendered words and uses a statistical hypothesis test to assess the statistical significance of the difference. Later, WEAT was used for measuring other bias types (e.g. Race) as well. Caliskan et al. (2017) in fact use effect sizes as a metric, but this was not taken up by follow-up work in NLP such as Gonen and Goldberg (2019).

Going beyond gender, Garg et al. (2018) analyzed ethnic biases in historical embeddings covering 100 years of language use. Swinger et al. (2019) showed that word embeddings of names reflect broad societal biases that are associated with those names, including race, gender, and age biases. Comparable biases also have been demonstrated in multilingual embeddings (Lauscher and Glavaš, 2019; Zhao et al., 2020). The perspective on types and sources of bias is continuing to broaden; Hovy and Prabhumoye (2021) propose a taxonomy of five sources of bias in NLP systems, namely the data, the annotation process, the input representations, the models, and the research design.

Bias in NLP systems. At the system level, bias has been investigated in applications including named entity recognition (NER), Machine Translation (MT), Sentiment Analysis, and Coreference Resolution. Kiritchenko and Mohammad (2018) examined 219 sentiment analysis systems and found that a majority exhibits gender and race biases. Mehrabi et al. (2019) reported that NER models recognize male names with higher recall compared to female names. Rudinger et al. (2018) and Zhao et al. (2018) showed that coreference resolution systems perform unequally across gender groups by associating occupations (such as doctor and engineer) more with men and others (like nurse) more with women. Similarly, Stanovsky et al. (2019) found that both commercial and academic MT models are at risk of generating translations based on gender stereotypes rather than the actual source content.

Bias in systems is usually measured by using benchmarks datasets for specific tasks with a one-factor design which are created to be as balanced as possible while varying the levels of the bias variable. Examples include WinoBias (Zhao et al., 2018) and WinoGender (Rudinger et al., 2018), two benchmarks for gender bias in coreference resolution which contrast "pro-stereotype" cases (the correct antecedent of a pronoun is conventionally associated with the pronoun's gender) and "anti-stereotype" cases (opposite situation); GAP (Webster et al., 2018), a dataset for the same task described in detail in Section 5; and the Equity Evaluation Corpus (EEC, Kiritchenko and Mohammad (2018)), developed to analyze gender and race bias in sentiment analysis and described in detail in Section 4. Bias is then quantified by measuring the differences in performance between these levels. Sometimes, but not always, the differences are subsequently tested for statistical significance, e.g. t-tests. To our knowledge, almost no studies on system-level bias have considered covariates, nor computed effect sizes, which makes them vulnerable to the criticisms outlined in Section 1.

An exception is a recent study Feder et al. (2021) which, like ours, disentangles bias from confounding factors. However, instead of performing correlational

analysis of model predictions, they aim at full-fledged causal analysis. Since causal relations can often not be recovered from data (Pearl, 2009), they assume that a causal graph modeling dependencies between predictors are given by a domain expert and show how to fine-tune contextualized embedding models with adversarial training to minimize bias. Thus, the two studies take complementary approaches: Feder et al. (2021) applies to model construction, while our study carries out black-box analysis of existing models.

Bias Mitigation. There are two main families of methods to mitigate bias at the representation level. Approaches from the first family create a modified version of the original data set that is biased in the opposite direction, training models on the union (Park et al., 2018; Zhao et al., 2019; Stanovsky et al., 2019). Approaches from the second family mitigate bias by transforming learned embeddings according to some balancing objective (Lauscher and Glavaš, 2019; Kaneko and Bollegala, 2019; Dev et al., 2020; Kaneko and Bollegala, 2021a,b).

At the system level, Zhao et al. (2017) proposed to constrain model predictions to follow a distribution from a training corpus. Rather than constraining the output, some of the previous work such as Elazar and Goldberg (2018); Zhang et al. (2018) and Kumar et al. (2019) used adversarial learning to remove unintended bias from the latent space during model training. Adjusting the loss function is another popular system level approach for bias mitigation. For instance, Qian et al. (2019) introduces a new term to the loss function to equalize the probabilities of male and female words in the output, and Jin et al. (2021) introduce a regularization term which reduces the importance placed on surface patterns.

Note that almost all mitigation methods require knowledge about which variables are (potentially) introducing bias, underlining the importance of reliable identification of bias variables.

3 Bias Identification With Regression Models: A Workflow

Following the discussion in the previous sections, the task of ("in vitro") bias identification is to establish that a bias variable – in contrast to other covariates which act as confounders – is primarily responsible for systematic variance in an observed variable, namely the performance of some computer system.

This is, of course, a very general task that arises in many empirical fields. A prominent family of techniques to address this task is *matching* (Rubin, 1973), which aims at generating two datasets that differ in the bias variable, but are as close as possible in their distribution over the covariates, so that any difference between the two datasets can be attributed to the bias variable. Matching is widely used in social sciences, economy, and medicine and many specific methods exist; see Stuart (2010) for an overview.[1]

Importantly, matching takes place *a priori*, before the experiment is carried out. This poses two challenges for applications in natural language processing: (a), dataset creation is dependent on the selection of covariates, so that it is not possible to assess the impact of new covariates on existing datasets without loss of comparability; (b), matching samples from the set of all datapoints, creating controlled rather than natural datasets, which may conflict with the desideratum of estimating model performance in broad-coverage scenarios.

The alternative is to carry out a *post-hoc* analysis that assesses the effects of the various covariates. The intuition is to start from a simple pairwise comparison of two levels of a bias variable (cf. the first and second column in Table 1) and add covariates to see whether the effect of the bias variable remains unaffected. This procedure has become standard in the last decade in neighboring fields like linguistics and psychology which have moved from significance tests (Student's t-test, analysis of variance) to the family of *multivariate regression models* (Bresnan et al., 2007; Baayen, 2008; Jaeger, 2008; Snijders and Bosker, 2012). Regression models estimate the relationships between the dependent (previously called observed) variable – in this case, system performance – and one or more independent variables – in this case, the putative bias variable and its covariates, each of which is assigned a direction and a significance. Since dataset creation is dependent from covariate analysis, regression models can be used to test new candidates for confounders on existing datasets.

At this point, it can be whether the fundamentally linear regression models are the right tool for the job, in particular given the broad success of non-linear deep learning models in NLP over the last years. We believe that it makes sense to distinguish carefully between the task of *output prediction* (given language input, predict language output) on which non-linear models indeed excel and the task of *performance prediction* (given [meta data for an] input and a model, predict how well the model does on the input). The latter is a considerably simpler problem which permits the use of linear models, as evidenced by a number of successful studies taking this approach (Beinborn et al., 2014; Papay et al., 2020; Caucheteux and King, 2022).

This section provides a practical workflow to set up a regression model for bias analysis, shown in Figure 1. Our starting point is the presence of a dataset with system predictions. Step 1 is the selection of an appropriate regression model. In Step 2, we choose a set

[1] Note that the term *bias* is used differently in the matching literature, namely as the effect of confounders on the observed variable.

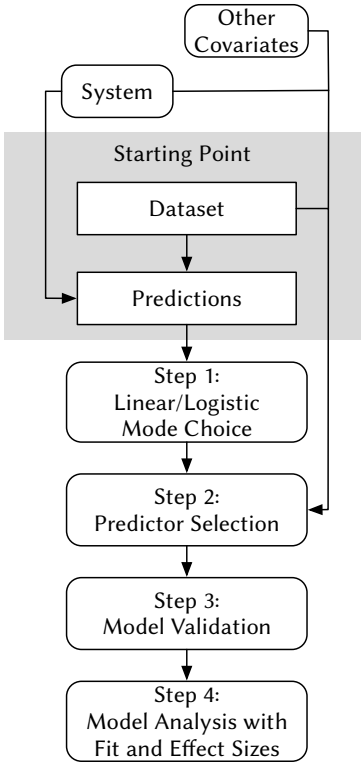


Figure 1: Workflow for regression-based bias analysis

of predictors with the potential to systematically influence the predictions of the systems, (i.e., the putative bias variable and plausible confounders) and carry out a regression analysis. Next, Step 3, model validation, ensures that the regression model is well specified and interpretable. Finally, Step 4 utilizes effect size analysis methods to explore how much of the system predictions can be attributed to the influence of the predictors.

Running example. We will illustrate the steps of the workflow on an actual (non-NLP) example, namely the effect of smoking on mortality, a topic of long-running interest in public health that has been analyzed extensively with regression models. The most basic finding is that smoking, overall, causes a strong increase in mortality (Doll et al., 2004). Why it is still reasonable to carry out a regression analysis in this case is that other lifestyle choices (alcohol consumption, diet, etc.) also presumably influence mortality, but exhibit correlations (Padrão et al., 2007). These are sometimes surprising – e.g., Tjønneland et al. (1999) found a correlation between wine and healthy diet. At the same time, approaches like matching are not applicable since the lifestyle properties of the participants cannot be influenced retroactively.

3.1 Step 1: Choice of Regression Model

The most common two forms of regression analysis are linear regression and logistic regression. When used to analyze the output of computational models, linear regression is appropriate to analyze the output of regression tasks, and logistic regression for the output of classification tasks.

Linear regression predicts the outcome of a continuous random variable y as a linear combination of weighted predictors x_i:

$$y \sim \alpha_1 x_1 + \cdots + \alpha_n x_n \tag{1}$$

where the coefficients α_i can be interpreted as the change in y resulting from a change in predictor x_i, keeping the other predictors constant.[2]

In contrast to linear regression, logistic regression does not model the outcome of the binary random variable y directly. Instead, it models the probability $P(y = 1)$, assuming that $P(y = 1)$ stands in a linear relationship to the logistically transformed linear combination of weighted predictors:

$$P(y = 1) \sim \sigma(\alpha_1 x_1 + \cdots + \alpha_n x_n) \tag{2}$$

where $\sigma(x) = 1/(1 + e^{-x})$ is the logistic function. Here, the coefficients α can be interpreted as the change in the logit for a unit change in the predictor.

Both types of regression support continuous, binary, and categorical predictors; the latter type is generally represented as a set of binary indicator predictors. As indicated above, these models assume that the predictors have an additive effect on the dependent variable (in the linear case) or its logit (in the logarithmic case).

Running example. In our mortality example, the outcome of the regression model is (some variant of) a death rate. Depending on the exact choice of measure, it might be appropriate to choose a linear regression model, when the death rates are approximately normally distributed (Gardner, 1973); or it might be appropriate to choose a logistic regression model, when the death rates can be interpreted as probabilities (Zhu et al., 2015b).

3.2 Step 2: Selection of Predictors

Maybe the most central step in the use of a regression model for bias analysis is the selection of the set of predictors for the regression model – that is, the putative bias variable and a set of plausible confounders to assess the respective roles of these variables in explaining the variance of the dependent variable.

[2] If the dependent variable is not (approximately) normally distributed, other types such as Poisson or negative binomial regression may be more appropriate.

This task is the responsibility of the user and typically involves domain knowledge. Typically, a user carrying out a bias identification analysis will have one (or a small number) of bias variables in mind, but need to select plausible confounders.

The five primary sources of bias variables given by Hovy and Prabhumoye (2021) can also serve as sources of confounders. The most straightforward of these are *data* and *input representations*, that is, properties of the text underlying the model, many of which are known to impact model performance. For example, low-frequency words and classes are modeled less reliably, longer stretches of text are harder to analyze, and so on (Poliak et al., 2018; Dayanik and Padó, 2020). Similarly, differences among *annotators* (age, social and cultural background, task familiarity) can impact model performance through labeling decisions (Sap et al., 2019), and obviously design decisions of the *system*, such as the choice of neural network architecture, contribute as well (Basta et al., 2019). Hovy and Prabhumoye's fifth category of *research design* is least relevant for our purposes, since it is concerned with systematic gaps in the field as such rather than analysis of individual studies.

Thus, for many problems, there will be a range of theoretically motivated covariates. The actual analysis will proceed in an interlocking fashion between exploratory data analysis based on domain knowledge – to identify interesting candidates for covariates – and regression modeling – to obtain statistically sound assessments of these covariates. In practical terms, the limiting factor is often that covariates need to be available as annotation on the dataset under consideration. While this is often relatively simple for the domains of input representation and systems, and doable for the domain of data, only recently has natural language processing started to record and analyze annotator properties (Sap et al., 2019), and there is an inherent tension between insights into annotation biases and annotator privacy. In some cases, however, covariates can be obtained by automatic or semi-automatic means. As an example, see our estimation of the typical age for the bearer of a specific first name on the basis of census data in Experiment 1 below. Such approaches can ease the burden of data collection, but the analysis should take into account the uncertainty introduced by automatic annotation.

Running example. In our lifestyle example, the covariates ideally include as many lifestyle factors as possible (such as alcohol consumption, diet, exercise, occupational hazards) as well as environmental factors (housing, climate) and personal factors such as family history of certain diseases. In practice, again, only a limited range of such factors is likely to be available.

3.3 Step 3: Model Validation

While regression models technically support arbitrary covariates, strong correlations among predictors, so-called *multicollinearity*, can distort the estimation of coefficients to the point that predictors are suggested to be significant when they are not, and vice versa (McNamee, 2005). Therefore, models should be checked for the presence of multicollinearity. There is a wide range of tests available, see Imdad Ullah et al. (2019) for a recent overview. We use the so-called variance inflation factor (VIF). VIF measures how much the variance of a predictor's coefficient is inflated due to correlations with other predictors. The VIF is computed for each independent variable V_i as

$$\mathrm{VIF_i} = 1/\left(1 - \mathrm{R_i^2}\right) \tag{3}$$

where R_i^2 is the correlation coefficient obtained when predicting α_i from all other predictors. Thus, the more collinearity is present, the higher VIF_i. VIF values of 4 or greater indicate severe multicollinearity, and values above 2.5 call for further investigation (Salmerón et al., 2018). In this case, a number of strategies are available, including dropping covariates, dimensionality reduction, and regularization methods (see Dormann et al. (2013) for details).

Another possible component of model validation is *predictor (feature) selection* based on an analysis of feature contributions. In many NLP tasks, irrelevant or unimportant features are removed for reasons of efficiency or to avoid overfitting (Li et al., 2009). In fields like psychology, where models serve explanatory purposes, predictor selection is discussed more controversially (Barr et al., 2013; Bates et al., 2018). In bias analysis, the goal is to test whether the effect of the putative bias variable stands up to the addition of covariates – the more covariates added to the model while retaining a significant contribution of the bias variable, the stronger the evidence for a specific role of the bias variable. For this reason, we believe that regression based bias analysis should be carried out on a comprehensive set of predictors, without feature selection (Barr et al., 2013).

Running example. In our lifestyle example, is it arguably important to check for multicollinearity, since the various covariates may be predictive of one another. For example, cramped housing conditions and occupational hazards are strongly linked through the shared cause of poverty (Hajat et al., 2015).

3.4 Step 4: Computing Model Fit and Effect Sizes

The coefficients α computed by regression models (cf. Step 1) are accompanied by indications of the confidence

level at which they are different from zero (i.e., whether the predictor has a significant effect). Furthermore, the global quality of regression models can be assessed by a number of statistics. Among them, we use *goodness of fit* which describes the proportion of the variance in the data that is explained by the independent variables of a regression model. The goodness of fit of a linear regression model is measured by R^2:

$$R^2 = \frac{\sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \tag{4}$$

where $\hat{y}_i$ is the model's prediction for data point i and $\bar{y}$ is the mean of the observations.

In logistic regression, there is no exact equivalent of R^2. Among several pseudo R^2 measures that have been proposed, Aldrich-Nelson pseudo-R^2 with Veall-Zimmermann correction (R^2_{VZ}) most closely approximates the R^2 in linear regression (Smith and Mckenna, 2013):

$$R^2_{\text{VZ}} = \frac{2[\text{LL(Null)} - \text{LL(Full)}]}{2[\text{LL(Null)} - \text{LL(Full)}] + N} \frac{2\text{LL(Null)} - N}{2\text{LL(Full)}} \tag{5}$$

where LL(Full) and LL(Null) are the log-likelihood values for the model with all predictors and for the empty model (without predictors), respectively.

Goodness of fit measures the overall ability of the model to explain the dependent variable. *Relative importance*, on the other hand, refers to the contribution of individual predictors (Achen, 1982). While assessment of relative importance in linear models with uncorrelated independent variables is simple (the impact of each predictor is its R^2 in univariate regression), in real-world datasets variables are generally correlated, as a result of which their impacts are not additive (Grömping, 2006). Lindeman-Merenda-Gold (LMG) scores (Lindeman et al., 1980) and Dominance Analysis (Budescu, 1993) are two popular techniques to figure out the individual contributions to the R^2 of the model of the predictors in linear and logistic regression, respectively.

The LMG method adds predictors to the regression model sequentially, and considers the resulting increase in R^2 as its contribution. Since this method depends on the possible orders in which predictors are added, the LMG score of a predictor x_k when added to a model with a set of predictors P is defined as the average of the increase in R^2 when adding x_k to all subsets of P (Grömping, 2006):

$$\text{seq}\, R^2(M|S) = R^2(M \cup S) - R^2(S) \tag{6}$$

$$\text{LMG}\,(x_k) = \frac{1}{n} \sum_{j=0}^{p-1} \sum_{\substack{S \subseteq P \\ n(S)=j}} \frac{\text{seq}\, R^2(\{x_k\}|S)}{\binom{p-1}{j}} \tag{7}$$

where $R^2(S)$ corresponds to the goodness of fit measure of a model with regressors in set S (cf. Eq 1) and $\text{seq}\, R^2(M|S)$ refers to the increase in R^2 when the regressors from M are added to the model based on the regressors S.

For logistic regression, there is again no direct counterpart. We propose Dominance Analysis (Budescu, 1993) as a measure of the relative importance of each predictor. Dominance analysis considers one predictor (x_i) to completely dominate another (x_j) if x_i's additional contribution to every possible model which does not include these two predictors is greater than contribution of x_j. In cases where complete dominance cannot be established, general dominance can also be used. One predictor generally dominates another if its average conditional contribution over all model sizes is greater than that of the other predictors (Azen and Traxel, 2009).

We propose the following interpretations for the regression scores outlined above: (a) At the system level, R^2 and pseudo-R^2 are indicators of the amount of variance in the system predictions that can be explained by the predictors and measure the *systematic bias* of a system. (b) At the predictor level, the significance of a predictor indicates the *presence of a specific bias*, and its effect size measures its *practical impact*; (c) the sign of a coefficient indicates the *direction* of a bias.

Regarding (b), an important difference between the application of significance testing in bias analysis and the usual use in NLP to compare competing models is that in our case, null results are arguably informative: they indicate the *absence* of a particular bias, according to the standards of significance. Naturally, the usual disclaimers regarding null results apply: care should be taken to ensure that they are not the result of faults in the experimental setup.

Running example. In our lifestyle example, the outcome of this step is a better understanding of individual risk factors, such as smoking, as opposed to the cluster of 'smoking and associated factors' that is obtained from a simple smoker-vs.-non-smoker analysis. Such an understanding is crucial to better assess the risk of individual patients based on their individual risk profile which might include compounding factors (high blood pressure, alcohol consumption) or mitigating factors (exercise, healthy diet). Again, note that the goal of this analysis is not to detract from the hazardous nature of smoking, but to better estimate of the effects of the relevant predictors on the outcome, namely mortality.

4 Experiment 1: Emotion Intensity Prediction

We now employ regression models to reanalyze model predictions on two experiments on standard datasets from the bias literature using the workflow defined in

Template
1. [PER] feels [EMO].
2. The situation makes [person] feel [EMO].
3. I made [person] feel [EMO].
4. [PER] made me feel [EMO].
5. [PER] found herself in a [EMO] situation.
6. [PER] told us about the recent [EMO] events.
7. The conversation with [person] was [EMO].
8. I saw [person] in the market.
9. I talked to [person] yesterday.
10. [PER] goes to school in our neighborhood.
11. [PER] has two children.

African American		European American	
Female	**Male**	**Female**	**Male**
Ebony	Alonzo	Amanda	Adam
Jasmine	Alphonse	Betsy	Alan
Lakisha	Darnell	Courtney	Andrew
Latisha	Jamel	Ellen	Frank
Latoya	Jerome	Heather	Harry
Nichelle	Lamar	Katie	Jack
Shaniqua	Leroy	Kristin	Josh
Shereen	Malik	Melanie	Justin
Tanisha	Terrence	Nancy	Roger
Tia	Torrance	Stephanie	Ryan

Table 2: Sentence templates in EEC dataset (top) and female and male first names associated with being African American and European American (bottom). [EMO]: an emotion adjective

Section 3.

Our first experiment is concerned with emotion intensity prediction. This task aims at combining discrete emotion classes with different levels of activation. Given a tweet and an emotion, the task requires to determine a score between 0 and 1 which is the intensity expressed regarding an emotion. Emotion intensity prediction was among the first NLP tasks to receive attention from a bias angle, when Kiritchenko and Mohammad (2018) found that among more than 200 emotion intensity prediction systems, almost all were biased with regard to gender or race. (In the remainder of the article, we will use 'system' to refer to models performing the task at hand, and 'model' to refer to the regression models we use for analyzing the systems' performance.)

4.1 Dataset and Previous Analysis

We use EEC, the same dataset used for the large-scale bias analysis of sentiment analysis mentioned above (Kiritchenko and Mohammad, 2018). EEC is a bias analysis benchmark created to evaluate fairness in sentiment analysis systems. It consists of 11 sentence templates

	train	dev	test	task
EI-reg	1701	388	1002	EIP
EEC	-	-	2100	EIP
GAP	-	2000	2000	CR

Table 3: Number of examples in the datasets used in our emotion intensity prediction (EIP) and coreference resolution (CR) experiments.

instantiated into 8,640 English sentences for four emotions (anger, joy, fear, sadness). Instantiated templates differ only in the name. [3] The dataset compares (a) male vs. female first names, and (b) European American vs. African American first names, using ten names of each category. Table 2 shows examples of such template sentences along with names that tend to belong to African American or European American demographic groups.

Kiritchenko and Mohammad (2018) used the EEC as a secondary test set for systems submitted to the SemEval 2018 Task 1 (Mohammad et al., 2018). For each system, they compared the average emotion intensities across different demographic groups using t-tests. They found that almost all systems consistently scored sentences of one gender and race higher than another, but bias directions were not consistent: e.g., some systems assigned higher emotion intensities to African Americans and lower ones to European Americans, while others show the opposite behavior. This apparently random behavior of the systems has no clear explanation and arguably raises concerns about a possible role of randomness in the analysis.

4.2 Systems

Since the predictions of the systems that participated in SemEval 2018 Task 1 are not publicly available[4], we instead implement and analyze five systems ourselves. Four systems represent the main architectures submitted to the shared task (Kiritchenko and Mohammad, 2018): A SVM unigram baseline and three neural systems based on word2vec word embeddings. To extend the model set to the current state of the art (2021), we include a transformer-based architecture as fifth system.

Support Vector Machine (SVM) We implement the unigram-based SVM used as baseline system in Mohammad et al. (2018).

[3] The EEC templates can also be instantiated using gendered noun phrases, but since these are unspecific with regard to the race variable, we focus on the version with proper nouns. This corresponds to the race analysis of the original study.

[4] Personal communication with the authors of shared task.

Convolutional Neural Network (CNN) Based on Aono and Himeno (2018), this system predicts an intensity score by first performing convolutions of different sizes on input word embeddings, followed by max-pooling and a shallow multi-layer perceptron (MLP).

Recurrent Neural Network (RNN) Our RNN is comparable to Wang and Zhou (2018). A two-layer BiLSTM traverses the input. The final hidden states in both directions from the final layer are concatenated and fed to a fully connected layer.

Attention Network (ATTN) This system is based on a CNN-LSTM architecture with attention similar to Wu et al. (2018). The input is fed to a single-layer BiLSTM. Next, an attention mechanism weights the hidden states, which are then passed through a CNN. The outputs of the CNN feature maps are concatenated and passed through a pooling layer and two fully connected layers.

Transformer-Based Neural Network (BERT) This system is based on the BERT_{BASE} multilayer bidirectional Transformer architecture (Devlin et al., 2019). It adds a linear layer on top of BERT and uses the final hidden state of the special [CLS] token as the latent representation of the input tweet, inspired by May et al. (2019).

We train and evaluate all the systems on the Anger partition of the EI-reg corpus (Mohammad and Bravo-Marquez, 2017) and EEC respectively. EI-reg was created by querying tweets in three languages (English, Arabic, Spanish) and for four emotions (Anger, Fear, Joy, Sadness) with words that were associated with the emotion at different intensity levels, such as *angry, annoyed, irritated* for Anger. Table 3 shows data statistics for both datasets.

4.3 Setup of the Regression Model

Bias Variable. In the EEC setup, the input sentences differ only in the person names that are filled in. We use the same two bias variables considered by the original study, namely Race and Gender.

Covariates. Due to the minimalist nature of the templates, coupled with the fact that the only part of the templates that is manipulated across conditions is the names, there is a limited range of linguistic properties that can systematically covary with bias. We consider two that we consider promising candidates. The first one is the (perceived) Age of a name is computed as the mean age for each name from US Social Security data.[5]

[5]We use data from `https://bit.ly/34cgjki` and the methodology from `https://bit.ly/30f8lps`.

Example	Properties				Intensity
	Gender	Race	Age	Freq	
Frank feels angry	Male	EA.	Old	0.05	0.55
Alonzo feels angry	Male	AA.	Old	0.24	0.48
Justin feels angry	Male	EA.	Yng	0.27	0.46
Lamar feels angry	Male	AA.	Yng	0.42	0.49
Jasmine feels angry	Female	AA.	Yng	0.47	0.47
Ellen feels angry	Female	EA.	Old	0.19	0.50

Table 4: Example sentences for the first template from Table 2 with their properties (EA.: European American, AA.: African American, Yng: Young). Intensity predicted by the the RNN system.

We discretize age, using 40 as the young/old boundary, following the assumption that 'older' names occur in different contexts than 'younger' names. The second covariate is the linguistic frequency of the name in the training data, since low-frequency names have found to be a source of low performance in NLP models (Dayanik and Padó, 2020). Since no explicit frequencies are available for the Google News skipgram vectors (Mikolov et al., 2013), we approximate frequency by vector length, which correlates highly with frequency (Roller and Erk, 2016). This is different from the 'real world' frequency of the name, which arguably is less likely to reflect in the behavior of an NLP model. Table 4 shows examples from the EEC with their properties.[6]

Model Shape We analyze the intensities predicted by our systems as in the original study, performing linear regression analysis at the level of each template with the following model:

$$\text{Intensity} \sim \text{Race} + \text{Gender} + \text{Age} + \text{Freq} \quad (8)$$

For Race, 1 means African American and 0 European American. For Gender, 1 means male and 0 female. For Age, 1 means young and 0 old.

Recall that on this task, there is no right or wrong answers. Instead, the focus of interest is whether the systems assign different intensities to a template dependent on the properties of the instantiating name. If they do not, none of the predictors will show a significant effect; if they do, significant effects will emerge.

Model Validation. Table 5 shows the variance inflation factors for the variables. Since only a single VIF value is larger than 2.5, and only marginally so, we conclude that multicollinearity is not a problem.

[6]We also performed experiments using a non-discretized version of age and including real-world frequency. We observed a substantially similar outcome (same levels of significance, coefficient signs for predictors, and almost the same overall R^2 values).

	Race	Gender	Frequency	Age
VIF	2.03	1.42	2.68	1.29

Table 5: VIF scores for the full set of variables.

		CNN	RNN	ATTN	BERT	SVM
R	Coef.	−0.010*	−0.010*	−0.002	−0.008	0.001
	Abs. LMG	0.080	0.082	0.010	0.068	0.018
	Per. LMG	0.42	0.47	0.06	0.48	0.03
G	Coef.	0.006	0.002	0.001	−0.001	−0.003***
	Abs. LMG	0.037	0.003	0.020	0.025	0.523
	Per. LMG	0.20	0.02	0.12	0.18	0.86
A	Coef.	0.005	0.001	0.001*	−0.003	0.001
	Abs. LMG	0.049	0.060	0.070	0.027	0.014
	Per. LMG	0.26	0.34	0.40	0.19	0.02
F	Coef.	0.016	0.019	0.015	0.010	−0.001
	Abs. LMG	0.023	0.029	0.073	0.021	0.048
	Per. LMG	0.12	0.17	0.42	0.15	0.08
R^2	model fit	0.19	0.17	0.17	0.14	0.60

Table 6: Regression-based bias analysis on EEC (R = Race, G = Gender, A = Age, F= Frequency) (Abs:Absolute, Per. Percentage)

4.4 Results

Table 6 shows the main results. (We omit intercepts in the table). The columns correspond to systems, and the rows describe the effects of bias variables for each system. For each predictor, we show a coefficient, a confidence level,[7] and an LMG effect size score.

Overall results As discussed in Section 3, we treat R^2 as a measure of systematic bias in a system. Inspection of the R^2 scores indicates that there is a certain amount of systematic bias in all systems, but that the three static-embedding neural systems do a very good job (R^2 between 0.17 and 0.19) compared to the SVM (R^2=0.60). BERT, the only neural system using contextualized embeddings, does an even better job and contains the least amount of systematic bias (R^2=0.14).

Comparison among systems None of the neural systems exhibits a significant gender bias, as the LMG scores show. Unlike Gender, the Race variable is responsible for the significant portion of the amount of variance in the system predictions. The CNN and the RNN systems both show a significant race bias which accounts of about 42–47% (LMG score: $\sim$ 0.08) of the variance in the intensity predictions. Note that Age, even though it misses significance, also accounts for 25–35% of the variation in intensity in the CNN and RNN. Interestingly, the ATTN architecture shows a different picture: there is a considerable amount of Age bias (40% of variance), but a much smaller race bias; instead, this system shows a frequency bias, which accounts for another 40% of the variance. In the BERT system, none of the bias variable achieve significance. In terms of relative contribution of individual predictors, BERT is more similar to CNN and RNN than to ATTN: Race is still making the largest contribution to the overall bias of the system, with 48%. The SVM differs strikingly: there are hardly any Race and Age biases, but an extremely strong effect of gender (86% of variance). Since this system does not use embeddings, the most likely source of this bias is the training corpus (EI-Reg), as also pointed out by the authors of the original study (Kiritchenko and Mohammad, 2018).

Interpretation While we can confirm the overall race bias found by Kiritchenko and Mohammad (2018), our picture differs substantially: (a) the direction of the bias is consistent among systems: all neural systems predict lower intensity scores for African Americans; (b) we do not observe a significant gender bias among neural systems; (c) we achieve a richer understanding of the systems' predictions, by quantifying the role of these factors, and by adding age and frequency into the picture.

Inspection of Examples Following up on (c), Table 4 presents three pairs of examples from the EEC dataset with their associated intensity values, as predicted by the RNN system. We have selected these instances to highlight the usefulness of the regression model to identify interesting instances. They show that the effect of Race variable (African Americans are assigned lower intensities) can be nullified by age (third example) and frequency (first and second examples). Such considerations remain hidden in an analysis that simply compares means between different groups of predictions.

5 Experiment 2: Coreference Resolution

Our second experiment analyzes several coreference resolvers in order to show how the logistic regression version of our approach can perform bias analysis on classification models. We choose coreference resolution as our task because of its established status in bias analysis; previous work has established that bias, in particular gender bias, is present in numerous coreference systems (Webster et al., 2018; Rudinger et al., 2018; Zhao et al., 2018). At the same time, coreference resolution, as a discourse level task, is faced with more complex data

[7] We use * for α=0.05, ** for α=0.01, and *** for α=0.001.

Input:
'He co-starred with Geena Davis in the TV show Sara, playing **her** next-door neighbor Stuart Webber.'

Person named entities:
Geena Davis (Correct), Sara (Incorrect)

Covariates:

	Geena Davis (Correct)	Sara (Incorrect)
Gender	1	1
Frequency	0.0015	0.0001
Diff	9	3
Single	0	1
Same	1	1

Figure 2: Example from the GAP dataset.

than more local (i.e., sentence-level) tasks, with a correspondingly larger set of potential confounders. We re-analyze a well-known coreference resolution dataset to verify the presence of gender bias in a manner that is robust against possible covariates.

5.1 Dataset and Previous Analysis

We use GAP (Webster et al., 2018), a human-labeled corpus of ambiguous pronoun-name pairs from English Wikipedia snippets. Each instance in the corpus contains two person named entities of the same gender and an ambiguous pronoun that may refer to either, or neither. System clusters were scored against GAP examples according to whether the cluster containing the target pronoun also contained the correct name (True Positive) or the incorrect name (False Positive). Figure 2 shows an example from the GAP development set (more statistics in Table 3).

In line with previous work (Webster et al., 2018), we use the development set of GAP to carry out our analyses. Below, we report overall system performance on the complete development set, in line with previous work. However, we exclude ≈200 instances from the development set, for which the pronoun does not refer to either of the two candidate named entities, from the regression analysis, since this makes it impossible to compute some of our covariates (cf. Section 5.3).

5.2 Systems

We experiment with six diverse coreference resolvers and analyze their predictions with our approach. As trained versions of all systems were publicly available, we did not need to train any systems ourselves. All systems except the BERT-based one were trained on the English portion of the 2012 CoNLL Shared Task dataset (Pradhan et al., 2012). It contains 2802 training, 343 development documents, and 348 test documents. BERT$_{large}$ Joshi et al. (2020) was pretrained on BooksCorpus (Zhu et al., 2015a) and English Wikipedia using cased *Wordpieces* tokens (Schuster and Nakajima, 2012) and fine-tuned on the 2012 CoNLL ST dataset.

Lee et al. (2013) This system is a collection of deterministic coreference resolution modules that incorporate lexical, syntactic, semantic, and discourse information, incorporating global document-level information. The system won the CoNLL 2011 shared task.

Clark and Manning (2015) This system uses a feature-rich machine learning approach. It performs entity clustering using the scores produced by two logistic classifier-based mention pair classifiers features. Both mention pair classifiers use a variety of common features such as syntactic, semantic and lexical features for mention pair classification.

Wiseman et al. (2016) This was the first neural coreference resolution system which showed that the task could benefit from modeling global features about entity clusters. It uses a neural mention ranker which is augmented by entity-level information produced by a RNN running over the cluster of candidate antecedents.

Lee et al. (2017) This was the first neural end-to-end coreference resolution system that works without a syntactic parser or hand engineered mention detector. It uses a combination of Glove and character level embeddings learnt by a CNN to represent the words of annotated documents. Next, the vectorized sentences of the document are fed into a BiLSTM to encode sentences and obtain span representations. The system also uses an attention mechanism to identify the head words in the span representations. Finally, the scoring functions are implemented via two feed-forward layers.

Lee et al. (2018) This system is an extension of Lee et al. (2017), which improves on two aspects. First, it uses gated attention mechanism which allows refinements in span representations; second, the system applies antecedent pruning which alleviates the complexity of running on long documents. It formed the state of the art for two years.

Joshi et al. (2020) SpanBERT is a variant of the BERT transformer (Devlin et al., 2019) designed to better represent spans of text. It works by (1) masking contiguous random spans, rather than random tokens, and (2) introducing a new objective function called span-boundary objective (SBO) which forces the model to learn to predict the entire masked span from the observed tokens at its boundary. BERT$_{large}$ trained with the SpanBERT

	Gender	C_Freq	C_Diff	C_Single	C_Same
VIF	1.03	1.03	1.88	1.02	1.53
	Gender	I_Freq	I_Diff	I_Single	I_Same
VIF	1.03	1.04	1.58	1.04	1.24

Table 7: VIF scores for the predictors. C_: Correct, I_: Incorrect

method improves the state of the art on many tasks including coreference resolution.

5.3 Setup of the Regression Model

Bias Variable. As in the original study, we use *Gender* as designated bias variable.

Covariates. In contrast to the first experiment, we do not use Age and Race, since the GAP dataset contains numerous named entities that are either not generally known or fictional (such as "the Hulk"). Therefore, these variables are either inapplicable or unknown to the typical annotator. Instead, use discourse-related properties of the antecedents as covariates, since in the task of coreference resolution the structural properties of the discourse arguably play a role in the difficulty of the task:

- *Diff* is the number of tokens between the named entity and target pronoun, normalized by the maximal distance in the corpus;
- *Single* states whether the named entity is a single word or an MWE;
- *Same* indicates whether the pronoun and named entity are in the same sentence;
- *Freq* defines the log-transformed corpus frequency of the entity, computed on the English Wikipedia (*en-wikipedia*) released on 20th March 2019, normalized by the maximal frequency in the corpus. The frequencies for MWEs are calculated based on the syntactic head of the expression.

Since the correct and the incorrect antecedent can differ regarding these properties, each property exists twice. We use the prefix C_for the correct and I_for the incorrect one. For gender, both antecedents have the same gender by design. The bottom part of Figure 2 shows how these covariates are initialized for the given example.

Model Shape We analyze the performance of the coreference resolvers at the level of individual predictions using following logistic regression model:

$$\begin{aligned} \text{p(Correct)} \sim \sigma(&\text{Gender} + \\ &\text{C\_Freq} + \text{I\_Freq} + \\ &\text{C\_Diff} + \text{I\_Diff} + \\ &\text{C\_Single} + \text{I\_Single} + \\ &\text{C\_Same} + \text{I\_Same}) \end{aligned} \tag{9}$$

where σ is the logistic function. p(Correct): is 1 if the resolver matches the pronoun with the correct named entity in corresponding instance and 0 otherwise. For Gender, 1 means female and 0 male. For Single, 1 means the entity is a single word, 0 otherwise. For Same, 1 means the entity is in the same sentence as the pronoun, 0 otherwise. We use Dominance Analysis to determine relative importance of each predictor.

In this setup, the regression model predicts whether each of the system predictions is correct or incorrect. To the extent the correctness is affected by the properties of the discourse captured by our predictors, we will obtain significant effects; conversely, should the correctness be fully random or dependent on properties independent from our predictors, we will not see significant effects.

Model Validation Table 7 shows the results of multicollinearity analysis on the set of predictors. All VIF values are smaller than 2, which indicates the absence of problematic multicollinearity.

	Male	Female	All	Bias
Lee et al. (2013)	55.4	45.5	50.5	0.82
Clark & Manning (2015)	58.5	51.3	55.0	0.88
Wiseman et al. (2016)	68.4	59.9	64.2	0.88
Lee et al. (2017)	67.2	62.2	64.7	0.92
Lee et al. (2018)	75.9	72.1	74.0	0.95
Joshi et al. (2020)	89.9	87.8	88.8	0.98

Table 8: F_1-Scores of resolvers on the GAP development set (Bias=F_1 Female / F_1 Male)

5.4 Results

Table 8 shows the performance of six resolvers on the complete GAP development set (overall and separately for Male and Female). It probably does not come as a surprise that performance increases over time; it is positive to note, though, that the Bias decreases correspondingly.

Table 9 shows the main results of our regression analysis on the subset of the GAP development set with a correct solution (cf. Section 5.1), organized by columns (systems). Each row provides a regression coefficient with its confidence level as well as the relative importance score for the predictor, using Dominance Analysis

		Lee et al. (2013)	Clark and Manning (2015)	Wiseman et al. (2016)	Lee et al. (2017)	Lee et al. (2018)	Joshi et al. (2020)
Gender	Coef	-0.473^{***}	-0.308^{**}	-0.314^{**}	-0.271^{**}	-0.215^{*}	-0.084
	DA	0.008	0.004	0.004	0.003	0.002	0.000
C_Freq	Coef	0.004	0.018^{***}	-0.004	-0.003	-0.001	0.001
	DA	0.000	0.005	0.000	0.000	0.000	0.000
I_Freq	Coef	-0.003	-0.003	-0.004	-0.006	-0.003	0.003
	DA	0.000	0.000	0.000	0.001	0.000	0.000
C_Diff	Coef	1.291^{**}	-1.617^{***}	$-0.933^{\cdot}$	-0.337	0.608	-0.065
	DA	0.006	0.002	0.001	0.001	0.001	0.000
I_Diff	Coef	-1.027^{*}	1.444^{***}	-0.086	$-0.740^{\cdot}$	-0.510	-0.053
	DA	0.003	0.002	0.001	0.004	0.001	0.000
C_Single	Coef	0.344^{**}	0.475^{***}	0.775^{***}	0.666^{***}	0.554^{***}	0.171
	DA	0.004	0.008	0.021	0.016	0.010	0.001
I_Single	Coef	-0.053	$-0.166^{\cdot}$	-0.268^{**}	-0.346^{***}	-0.360^{***}	0.036
	DA	0.001	0.001	0.003	0.006	0.006	0.000
C_Same	Coef	-0.603^{***}	-0.456^{***}	-0.561^{***}	-0.564^{***}	-0.336^{*}	-0.007
	DA	0.015	0.002	0.007	0.008	0.004	0.000
I_Same	Coef	0.086	0.366^{**}	0.120	0.318^{**}	0.317^{**}	0.028
	DA	0.000	0.002	0.000	0.003	0.003	0.000
Model Fit	R^2_{VZ}	0.05	0.04	0.05	0.05	0.03	0.01
	Acc	0.61 (0.58)	0.57 (0.55)	0.58 (0.53)	0.59 (0.53)	0.63 (0.63)	0.55 (0.55)

Table 9: Regression-based analysis of coreference resolution systems on GAP dataset.
DA: Dominance Analysis, Freq: Frequency, C_: Correct, I_: Incorrect instances.

(DA). R^2_{VZ} indicates the goodness of fit values at the level of complete systems. (Note that these numbers, computed for logistic regression models, are not comparable to the numbers for linear regression models from Experiment 1.)

We also report accuracy values for the predictions of our logistic regression model, averaged over 10-fold cross-validation (*Acc*). Numbers in parentheses indicate the accuracy of corresponding majority baselines. The differences in baseline scores across systems are due to the fact that gold labels (i.e., the p(Correct) variable in the equation) are dependent on system predictions.

System level analysis We first discuss results at the system level. The last row of Table 9 (Model Fit) shows the overall model fit for all systems. The ability of our regression model to outperform majority baselines for the first four systems (Lee et al., 2013; Clark and Manning, 2015; Wiseman et al., 2016; Lee et al., 2017) shows that our analysis can predict mistakes made by these coreference resolvers by only considering a small set of discourse-related features plus Gender. In contrast, Lee et al. (2018) and Joshi et al. (2020) both show an R^2_{VZ} of almost zero, that is, the logistic regression models perform at the level of a majority class baseline – the remaining errors that they systems make are idiosyncratic rather than systematic. These findings tie in well with the overall system performance scores shown in Table 8.

It is striking that Joshi et al. (2020), the best model by a substantial margin, is also the one exhibiting the smallest bias. We see two possible explanations: (a), the model was trained on a large corpus from several domains with different discourse style, which may make it more robust to gender bias (Saunders and Byrne, 2020); (b) in contrast to the older studies, this model is based on contextualized embeddings, which also showed lower bias in Experiment 1. Without re-training the model, we cannot currently distinguish between these two explanations.

Predictor level analysis We now move on to investigate the contribution of each predictor to the systems' predictability. At this level, gender is a statistically significant predictor ($p < 0.05$) for all systems except Joshi et al. (2020). It has a negative sign throughout, indicating worse performance for female entities. This is again in line with the findings reported in Table 8. However, our approach reveals other important patterns which cannot be observed by using traditional analysis methods. First, Clark and Manning (2015) and Wiseman et al. (2016) have the same DA coefficient for gender variable but different R^2_{VZ} values. We interpret this to mean that the contribution of gender bias to the overall bias in

these two systems is not the same, an observation that would not have been possible through traditional bias analysis methods (cf. Table 8).

Second, we see that the coefficient signs of the predictors C_Single and C_Same remain the same across systems: Systems perform better for instances where the correct antecedent is a single word, and it is not in the same sentence with the pronoun. Moreover, dominance analysis shows that these two predictors are among the main contributors to the biased predictions in four systems out of six, the two exceptions being Lee et al. (2013) and Joshi et al. (2020).

Third, the small but consistent positive relative importance values of the C_Diff and I_Diff predictors for half of the systems show that these variables help explain the systems' predictions. In contrast, the low relative importance values of the C_Frequency and I_Frequency predictors indicate that these variables do not affect coreference resolution much.

Interpretability These detailed findings indicate that, similar to emotion intensity prediction, the analysis of coreference resolvers can also benefit from not only the controlled bias variable but also from other properties of the input even in datasets which are designed carefully to isolate the effect of the target variable. As stated in Exp. 1, these analyses can also be used to extract interesting examples and subsets.

We illustrate this for the two attributes C_Same and I_Same, i.e., whether the correct and incorrect antecedent are in the same sentence or not. We split the GAP dataset into four reasonably-sized subsets based on the values of these attributes: the subset where both are in the same sentence (C_Same=1 and I_Same=1) includes ∼ 900 examples and the other three subsets include ∼ 300 examples. Table 10 shows the bias values (defined as above) for the three best performing systems. We observe that these systems vary widely regarding the subset where gender bias is most prominently visible varies across systems: Lee et al. (2017, 2018) both show the worst bias when the incorrect antecedent is not in the current sentence (I_Same=0), but differ in the effect of the position of the correct antecedent (C_Same). In contrast, Joshi et al. (2020) performs almost perfectly when I_Same=0, but struggles most the case when both correct and incorrect antecedent are in the current sentence. These variations in model performance across subsets raise questions about the representations of antecedents in the various models which go beyond the scope of this paper.

		I_Same=0	I_Same=1
Lee et al. (2017)	C_Same=0	**0.80**	1.10
	C_Same=1	0.90	0.90
		I_Same=0	I_Same=1
Lee et al. (2018)	C_Same=0	0.90	1.00
	C_Same=1	**0.86**	0.97
		I_Same=0	I_Same=1
Joshi et al. (2020)	C_Same=0	1.02	1.02
	C_Same=1	0.99	**0.94**

Table 10: Bias values for the three best performing systems, with data split into four groups according to C_Same and I_Same (worst bias marked in boldface).

6 Conclusion

In this article, we have argued that bias analysis, a task of major importance concerning the societal implications of NLP, can benefit from richer statistical methods to detect, quantify and attribute bias. We have proposed to follow other scientific fields in adopting regression analysis which (a) generalizes to multiple bias variables, (b) can quantify the contribution of confounder variables to the observed bias with measures of effect size, and (c) can be used to diagnose system behavior and extract informative datapoints.

Clearly, regression analysis is no panacea on its own: it presupposes a set of plausible covariates of bias, which can come from a wide variety of sources, including task-specific annotation, task-unspecific input representations, or model architecture (Hovy and Prabhumoye, 2021). Such covariates are typically known through domain expertise or uncovered by exploratory data analysis. Furthermore, the values of these bias variables must be available, or annotated, for all data points, which can represent a bottleneck. Thus, regression analysis complements, but does not replace, traditional methods of bias analysis.

We have demonstrated the usefulness of our approach by analyzing a range of model architectures on a regression task and a classification task, obtaining model-level results that are in line with the existing literature, e.g., BERT-based systems appear to exhibit comparatively little bias (Basta et al., 2019). In addition, adding predictor-level analysis offers a richer understanding of the importance of the bias variables and their interactions with other textual properties. Note that we only considered datasets specifically designed to exhibit the effects of a single bias variable. We believe that the benefits of our analysis framework would be even clearer on more naturalistic datasets where pairwise hypothesis tests become even more problematic (see, e.g., Gorrostieta et al. (2019)).

Another methodological debate that we hope to contribute to is what constitutes a 'substantial' bias? We have argued that effect sizes offer a statistically sound approach to measuring the amount of variation in the

output that can be attributes to a set of input properties. Our study provides a starting point for the community to establish a magnitude for what it considers a 'substantial' bias, similar to the often-used thresholds for inter-annotator agreement (Cohen, 1968) or general effect sizes in psychology (Cohen, 1988).

Regarding future work, one avenue concerns richer regression models that analyze interactions among predictors. Such interactions, when properly motivated, can further improve our understanding of the performance data. In fact, our last example in Exp. 2 essentially demonstrates an interaction: the degree of gender bias in the conference resolvers is affected by an interaction between the position of the incorrect and the correct antecedents. Ideally, such observations might serve as motivation for assessing and potentially modifying model architectures or training regimens.

Another avenue of future research is widening our scope from the analysis of bias in NLP models (that is, "in vitro" bias according to our terminology in Section 1) to real life "in vivo" bias in academic communities. Recent studies have identified multiple such biases, e.g., gender bias in publications (Mohammad, 2020) and hiring (Eaton et al., 2020). We would hope that the application of robust regression analysis, a standard method in the social sciences, would help bolstering these studies and contribute towards redressing such social harms.

References

Achen, Christopher H. 1982. *Interpreting and using regression*, volume 29 of *Quantitative Applications in the Social Sciences*. Sage.

Aono, Masaki and Shinnosuke Himeno. 2018. KDE-AFFECT at SemEval-2018 Task 1: Estimation of affects in tweet by using convolutional neural network for n-gram. In *Proceedings of SemEval*, pages 156–161, New Orleans, LA.

Azen, Razia and Nicole Traxel. 2009. Using dominance analysis to determine predictor importance in logistic regression. *Journal of Educational and Behavioral Statistics*, 34:319 –347.

Baayen, Harald. 2008. *Analyzing Linguistic Data*. Cambridge University Press.

Barr, Dale J, Roger Levy, Christoph Scheepers, and Harry J Tily. 2013. Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of memory and language*, 68(3):255–278.

Basta, Christine, Marta R. Costa-jussà, and Noe Casas. 2019. Evaluating the underlying gender bias in contextualized word embeddings. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 33–39, Florence, Italy. Association for Computational Linguistics.

Bates, Douglas, Reinhold Kliegl, Shravan Vasishth, and Harald Baayen. 2018. Parsimonious mixed models. ArXiv preprint, `http://arxiv.org/abs/1506.04967`.

Beinborn, Lisa, Torsten Zesch, and Iryna Gurevych. 2014. Predicting the difficulty of language proficiency tests. *Transactions of the Association for Computational Linguistics*, 2:517–530.

Bender, Emily M and Batya Friedman. 2018. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6:587–604.

Blodgett, Su Lin, Solon Barocas, Hal Daumé III, and Hanna Wallach. 2020. Language (technology) is power: A critical survey of "bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, Online. Association for Computational Linguistics.

Bolukbasi, Tolga, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam Tauman Kalai. 2016. Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. In *Proceedings of NeurIPS*, pages 4349–4357.

Bresnan, Joan, Anna Cueni, Tatiana Nikitina, and Harald Baayen. 2007. Predicting the dative alternation. In G. Bouma, I. Kraemer, and J. Zwarts, editors, *Cognitive Foundations of Interpretation*, pages 69–94. Royal Netherlands Academy of Science.

Budescu, David V. 1993. Dominance analysis: a new approach to the problem of relative importance of predictors in multiple regression. *Psychological bulletin*, 114(3):542.

Caliskan, Aylin, Joanna J Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186.

Caucheteux, Charlotte and Jean-Rémi King. 2022. Brains and algorithms partially converge in natural language processing. *Communications Biology*, 5(1):134.

Clark, Kevin and Christopher D. Manning. 2015. Entity-centric coreference resolution with model stacking. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1405–1415, Beijing, China. Association for Computational Linguistics.

Cohen, Jacob. 1968. Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit. *Psychological Bulletin*, 70:213–220.

Cohen, Jacob. 1988. *Statistical power analysis for the behavioral sciences*, 2nd edition. Lawrence Erlbaum Associates, Hillsdale, NJ.

Davidson, Thomas, Debasmita Bhattacharya, and Ingmar Weber. 2019. Racial bias in hate speech and abusive language detection datasets. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 25–35, Florence, Italy. Association for Computational Linguistics.

Dayanik, Erenay and Sebastian Padó. 2020. Masking actor information leads to fairer political claims detection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4385–4391, Online. Association for Computational Linguistics.

Dev, Sunipa, Tao Li, Jeff M. Phillips, and Vivek Srikumar. 2020. On measuring and mitigating biased inferences of word embeddings. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7659–7666. AAAI Press.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Díaz, Mark, Isaac Johnson, Amanda Lazar, Anne Marie Piper, and Darren Gergle. 2018. Addressing age-related bias in sentiment analysis. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pages 1–14.

Doll, Richard, Richard Peto, Jillian Boreham, and Isabelle Sutherland. 2004. Mortality in relation to smoking: 50 years' observations on male british doctors. *BMJ*, 328(7455):1519.

Dormann, Carsten, Jane Elith, Sven Bacher, Carsten Buchmann, Gudrun Carl, Gabriel Carré, T. Dieköt-ter, Jaime García Márquez, Bernd Gruber, Bruno Lafourcade, Pedro Leitão, Tamara Münkemüller, Colin Mcclean, Patrick Osborne, Björn Reineking, Boris Schröder, Andrew Skidmore, Damaris Zurell, and Sven Lautenbach. 2013. Collinearity: A review of methods to deal with it and a simulation study evaluating their performance. *Ecography*, 36:27–46.

Eaton, Asia A., Jessica F. Saunders, Ryan K. Jacobson, and Keon West. 2020. How gender and race stereotypes impact the advancement of scholars in stem: Professors'biased evaluations of physics and biology post-doctoral candidates. *Sex Roles*, 82(3):127–141.

Elazar, Yanai 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.

Feder, Amir, Nadav Oved, Uri Shalit, and Roi Reichart. 2021. CausaLM: Causal model explanation through counterfactual language models. *Computational Linguistics*, 47(2):333–386.

Friedman, Batya and Helen Nissenbaum. 1996. Bias in computer systems. *ACM Transactions on Information Systems*, 14(3):330–347.

Gao, Yang, Steffen Eger, Ilia Kuznetsov, Iryna Gurevych, and Yusuke Miyao. 2019. Does my rebuttal matter? insights from a major NLP conference. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1274–1290, Minneapolis, Minnesota. Association for Computational Linguistics.

Gardner, M. J. 1973. Using the environment to explain and predict mortality. *Journal of the Royal Statistical Society. Series A (General)*, 136(3):421–440.

Garg, Nikhil, Londa Schiebinger, Dan Jurafsky, and James Zou. 2018. Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences*, 115(16):E3635–E3644.

Gerz, Daniela, Ivan Vulić, Edoardo Maria Ponti, Roi Reichart, and Anna Korhonen. 2018. On the relation between linguistic typology and (limitations of) multilingual language modeling. In *Proceedings of EMNLP*, pages 316–327, Brussels, Belgium.

Gonen, Hila and Yoav Goldberg. 2019. Lipstick on a pig: Debiasing methods cover up systematic gender biases in word embeddings but do not remove them. In *Proceedings of the 2019 Workshop on Widening NLP*, pages 60–63, Florence, Italy. Association for Computational Linguistics.

Gorrostieta, Cristina, Reza Lotfian, Kye Taylor, Richard Brutti, and John Kane. 2019. Gender de-biasing in

speech emotion recognition. In *Proceedings of Interspeech*, pages 2823–2827.

Grömping, Ulrike. 2006. Relative importance for linear regression in R: the package `relaimpo`. *Journal of statistical software*, 17(1):1–27.

Hajat, Anjum, Charlene Hsia, and Marie S O'Neill. 2015. Socioeconomic disparities and air pollution exposure: a global review. *Current Environmental Health Reports*, 2(4):440–450.

Hovy, Dirk and Shrimai Prabhumoye. 2021. Five sources of bias in natural language processing. *Language and Linguistics Compass*, 15(8):e12432.

Imdad Ullah, Muhammad, Muhammad Aslam, Saima Altaf, and Munir Ahmed. 2019. Some new diagnostics of multicollinearity in linear regression model. *Sains Malaysiana*, 48(2):2051–2060.

Jaeger, T. Florian. 2008. Categorical data analysis: Away from anovas (transformation or not) and towards logit mixed models. *Journal of Memory and Language*, 59(4):434–446. Special Issue: Emerging Data Analysis.

Jin, Xisen, Francesco Barbieri, Brendan Kennedy, Aida Mostafazadeh Davani, Leonardo Neves, and Xiang Ren. 2021. On transferability of bias mitigation effects in language model fine-tuning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3770–3783, Online. Association for Computational Linguistics.

Joshi, Mandar, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Kaneko, Masahiro and Danushka Bollegala. 2019. Gender-preserving debiasing for pre-trained word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1641–1650, Florence, Italy. Association for Computational Linguistics.

Kaneko, Masahiro and Danushka Bollegala. 2021a. Debiasing pre-trained contextualised embeddings. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1256–1266, Online. Association for Computational Linguistics.

Kaneko, Masahiro and Danushka Bollegala. 2021b. Dictionary-based debiasing of pre-trained word embeddings. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 212–223, Online. Association for Computational Linguistics.

Kiritchenko, Svetlana and Saif Mohammad. 2018. Examining gender and race bias in two hundred sentiment analysis systems. In *Proceedings of STARSEM*, pages 43–53, New Orleans, LA.

Kumar, Sachin, Shuly Wintner, Noah A. Smith, and Yulia Tsvetkov. 2019. Topics to avoid: Demoting latent confounds in text classification. 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 4153–4163, Hong Kong, China. Association for Computational Linguistics.

Këpuska, V. and G. Bohouta. 2018. Next-generation of virtual personal assistants (Microsoft Cortana, Apple Siri, Amazon Alexa and Google Home). In *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 99–103.

Lauscher, Anne and Goran Glavaš. 2019. Are we consistently biased? multidimensional analysis of biases in distributional word vectors. In *Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (*SEM 2019)*, pages 85–91, Minneapolis, Minnesota. Association for Computational Linguistics.

Lee, Heeyoung, Angel Chang, Yves Peirsman, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2013. Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics*, 39(4):885–916.

Lee, Kenton, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark. Association for Computational Linguistics.

Lee, Kenton, Luheng He, and Luke Zettlemoyer. 2018. Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 687–692, New Orleans, Louisiana. Association for Computational Linguistics.

Li, Shoushan, Rui Xia, Chengqing Zong, and Chu-Ren Huang. 2009. A framework of feature selection methods for text categorization. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 692–700, Suntec, Singapore. Association for Computational Linguistics.

Lindeman, Richard H., Peter F. Merenda, and Ruth Z. Gold. 1980. *Introduction to Bivariate and Multivariate Analysis*. Scott Foresman, Glenview, IL, USA.

May, Chandler, Alex Wang, Shikha Bordia, Samuel R. Bowman, and Rachel Rudinger. 2019. On measuring social biases in sentence encoders. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 622–628, Minneapolis, Minnesota. Association for Computational Linguistics.

McNamee, Roseanne. 2005. Regression modelling and other methods to control confounding. *Occupational and Environmental Medicine*, 62(7):500–506.

Mehrabi, Ninareh, Thamme Gowda, Fred Morstatter, Nanyun Peng, and Aram Galstyan. 2019. Man is to person as woman is to location: Measuring gender bias in named entity recognition. In *Proceedings of the 31st ACM conference on Hypertext and Social Media*, pages 231–232.

Mehrabi, Ninareh, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. 2021. A survey on bias and fairness in machine learning. *ACM Computing Surveys*, 54(6).

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Proceedings of the 1st International Conference on Learning Representations (Workshop Track)*.

Mohammad, Saif, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. Semeval-2018 task 1: Affect in tweets. In *Proceedings of SEMEVAL*, pages 1–17, New Orleans, LA.

Mohammad, Saif M. 2020. Gender gap in natural language processing research: Disparities in authorship and citations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7860–7870, Online. Association for Computational Linguistics.

Mohammad, Saif M. and Felipe Bravo-Marquez. 2017. Emotion intensities in tweets. In *Proceedings of the sixth joint conference on lexical and computational semantics (*Sem)*, Vancouver, Canada.

Padrão, Patrícia, Nuno Lunet, Ana Cristina Santos, and Henrique Barros. 2007. Smoking, alcohol, and dietary choices: evidence from the Portuguese national health survey. *BMC Public Health*, 7(1):1–9.

Papay, Sean, Roman Klinger, and Sebastian Padó. 2020. Dissecting span identification tasks with performance prediction. In *Proceedings of EMNLP*, page 4881–4895, Online.

Park, Ji Ho, Jamin Shin, and Pascale Fung. 2018. Reducing gender bias in abusive language detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2799–2804, Brussels, Belgium. Association for Computational Linguistics.

Pearl, Judea. 2009. Causal inference in statistics: An overview. *Statistics Surveys*, 3:96–146.

Poliak, Adam, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191.

Pradhan, Sameer, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Joint Conference on EMNLP and CoNLL - Shared Task*, pages 1–40, Jeju Island, Korea. Association for Computational Linguistics.

Qian, Yusu, Urwa Muaz, Ben Zhang, and Jae Won Hyun. 2019. Reducing gender bias in word-level language models with a gender-equalizing loss function. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 223–228, Florence, Italy. Association for Computational Linguistics.

Roller, Stephen and Katrin Erk. 2016. PIC a different word: A simple model for lexical substitution in context. In *Proceedings of NAACL/HLT*, pages 1121–1126, San Diego, California.

Rubin, Donald B. 1973. Matching to remove bias in observational studies. *Biometrics*, pages 159–183.

Rudinger, Rachel, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. 2018. Gender bias in coreference resolution. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 8–14, New Orleans, LA.

Salmerón, R., C. B. García, and J. García. 2018. Variance inflation factor and condition number in multiple linear regression. *Journal of Statistical Computation and Simulation*, 88(12):2365–2384.

Sap, Maarten, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. 2019. The risk of racial bias in hate speech detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1668–1678, Florence, Italy. Association for Computational Linguistics.

Saunders, Danielle and Bill Byrne. 2020. Reducing gender bias in neural machine translation as a domain adaptation problem. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7724–7736, Online. Association for Computational Linguistics.

Schmid, Hans-Jörg. 2002. Do women and men really live in different cultures? Evidence from the BNC. In *Corpus linguistics by the Lune: a Festschrift for Geoffrey Leech*, pages 185–221. Peter Lang, Frankfurt.

Schuster, Mike and Kaisuke Nakajima. 2012. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5149–5152. IEEE.

Schwemmer, Carsten and Sebastian Jungkunz. 2019. Whose ideas are worth spreading? the representation of women and ethnic groups in ted talks. *Political Research Exchange*, 1(1):1–23.

Singh, Amit, Catherine Rose, Karthik Visweswariah, Vijil Chenthamarakshan, and Nandakishore Kambhatla. 2010. Prospect: A system for screening candidates for recruitment. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, page 659–668, New York, NY, USA. Association for Computing Machinery.

Smith, Thomas J. and C. M. Mckenna. 2013. A comparison of logistic regression Pseudo R^2 indices. *General Linear Model Journal*, 39(2):17–26.

Snijders, Tom and Roel Bosker. 2012. *Multilevel Analysis: An Introduction to Basic and Advanced Multilevel Modeling*, 2nd edition. Sage Publishers, London.

Stanovsky, Gabriel, Noah A. Smith, and Luke Zettlemoyer. 2019. Evaluating gender bias in machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1679–1684, Florence, Italy. Association for Computational Linguistics.

Stuart, Elizabeth A. 2010. Matching methods for causal inference: A review and a look forward. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 25(1):1.

Sullivan, Gail M. and R. Feinn. 2012. Using effect size – or why the *p* value is not enough. *Journal of graduate medical education*, 4(3):279–82.

Sun, Dongming, Xiaolu Zhang, Kim-Kwang Raymond Choo, Liang Hu, and Feng Wang. 2021. Nlp-based digital forensic investigation platform for online communications. *Computers & Security*, 104:102210.

Sun, Tony, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth Belding, Kai-Wei Chang, and William Yang Wang. 2019. Mitigating gender bias in natural language processing: Literature review. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1630–1640, Florence, Italy. Association for Computational Linguistics.

Swinger, Nathaniel, Maria De-Arteaga, Neil Thomas Heffernan IV, Mark DM Leiserson, and Adam Tauman Kalai. 2019. What are the biases in my word embedding? In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 305–311.

Tjønneland, Anne, Morten Grønbæk, Connie Stripp, and Kim Overvad. 1999. Wine intake and diet in a random sample of 48763 Danish men and women. *The American journal of clinical nutrition*, 69(1):49–54.

Wang, Min and Xiaobing Zhou. 2018. Yuan at SemEval-2018 Task 1: Tweets emotion intensity prediction using ensemble recurrent neural network. In *Proceedings of SEMEVAL*, pages 205–209, New Orleans, LA.

Webster, Kellie, Marta Recasens, Vera Axelrod, and Jason Baldridge. 2018. Mind the GAP: A balanced corpus of gendered ambiguous pronouns. *Transactions of the Association for Computational Linguistics*, 6:605–617.

Wiseman, Sam, Alexander M. Rush, and Stuart M. Shieber. 2016. Learning global features for coreference resolution. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 994–1004, San Diego, California. Association for Computational Linguistics.

Wu, Chuhan, Fangzhao Wu, Junxin Liu, Zhigang Yuan, Sixing Wu, and Yongfeng Huang. 2018. THU_NGN at SemEval-2018 Task 1: Fine-grained tweet sentiment intensity analysis with attention CNN-LSTM. In *Proceedings of SEMEVAL*, pages 186–192, New Orleans, Louisiana.

Wu, Yonghui, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. `https://arxiv.org/abs/1609.08144`.

Zhang, Brian Hu, Blake Lemoine, and Margaret Mitchell. 2018. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 335–340.

Zhao, Jieyu, Subhabrata Mukherjee, Saghar Hosseini, Kai-Wei Chang, and Ahmed Hassan Awadallah. 2020. Gender bias in multilingual embeddings and cross-lingual transfer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2896–2907, Online. Association for Computational Linguistics.

Zhao, Jieyu, Tianlu Wang, Mark Yatskar, Ryan Cotterell, Vicente Ordonez, and Kai-Wei Chang. 2019. Gender bias in contextualized word embeddings. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 629–634, Minneapolis, Minnesota.

Zhao, Jieyu, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2017. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2979–2989, Copenhagen, Denmark. Association for Computational Linguistics.

Zhao, Jieyu, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018. Gender bias in coreference resolution: Evaluation and debiasing methods. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 15–20, New Orleans, LA.

Zhu, Yukun, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015a. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

Zhu, Zhiwei, Zhi Li, David Wylde, Michael Failor, and George Hrischenko. 2015b. Logistic regression for insured mortality experience studies. *North American Actuarial Journal*, 19(4):241–255.

Policy-focused Stance Detection in Parliamentary Debate Speeches

Gavin Abercrombie, Department of Computer Science, University of Manchester
gavin.abercrombie@manchester.ac.uk
Riza Batista-Navarro, Department of Computer Science, University of Manchester
riza.batista@manchester.ac.uk

Abstract Legislative debate transcripts provide citizens with information about the activities of their elected representatives, but are difficult for people to process. We propose the task of policy-focused stance detection, in which both the policy proposals under debate and the position of the speakers towards those proposals are identified. We adapt a previously existing dataset to include manual annotations of *policy preferences*, an established schema from political science. We evaluate a range of approaches to the automatic classification of policy preferences and speech sentiment polarity, including transformer-based text representations and a multi-task learning paradigm. We find that it is possible to identify the policies under discussion using features derived from the speeches, and that incorporating motion-dependent debate modelling, previously used to classify speech sentiment, also improves performance in the classification of policy preferences. The proposed use of contextual embeddings and a multi-task learning paradigm do not perform as well as simpler approaches. We analyse the output of the best performing system, finding that discriminating features for the task are highly domain-specific, and that speeches that address policy preferences proposed by members of the same party can be among the most difficult to predict.

1 Introduction

Transcripts of legislative debates provide access to information concerning the policies that are publicly supported or opposed by politicians. They are of interest to political scientists, the media, the politicians themselves, and citizens who wish to monitor the activities of their representatives.

However, such documents are complex and difficult for people to process. Transcripts of debates in the United Kingdom (UK) Parliament are so hard for ordinary people to make sense of that parliamentary monitoring website `www.theyworkforyou.com` publishes manually annotated versions of the transcripts. These include crowd-sourced explanations of the debated proposals, as well as policy-focused aggregations of the voting records of parliamentarians. The large quantity and esoteric nature of the data in the parliamentary record (known as *Hansard*) motivates the need for automatic analysis of its contents.

Previous work in the domain of legislative debate transcripts has focused on either *(a)* sentiment polarity classification (Bhavan et al., 2019; Burfoot et al., 2011; Thomas et al., 2006), or *(b)* policy identification (Abercrombie and Batista-Navarro, 2018b; Abercrombie et al., 2019) in isolation. As far as we are aware, these two tasks have not previously been combined in this domain, despite the fact that: *(1)* the information yielded is complementary, and perhaps even necessary, for practical use (i.e., without analysis of debated policies, the target of sentiment in the speeches is unknown); and *(2)* these two tasks rely on features derived from shared information, which could assist with the learning of parameters for both tasks in a multi-task learning setting.

Borrowing the concept of *policy preferences* from political science, we compare approaches to automatically determining the policy preference that is under discussion in each debate, and whether each speaker supports or opposes it.

Our contributions Building on the work of Abercrombie et al. (2019); Abercrombie and Batista-Navarro (2020), we combine policy preference identification and speech-level sentiment polarity analysis to formulate the task of policy-focused speech stance detection for the domain of legislative debate speeches, in which the position of each speaker in a debate is identified in relation to the proposal under discussion. Unlike prior work, we thus obtain interpretable analysis of the positions taken by MPs with respect to the policies presented in parliamentary debates.

To this end, we add a set of manually annotated

policy preference labels to a large existing English language corpus of UK parliamentary debates, creating the first dataset to be labelled with both topics (policy preferences) and positions (sentiment) in this domain. We make the enhanced corpus available to the research community.

We use this dataset for the evaluation of approaches to the classification of policy-focused speaker stance. We test classification systems comprising combinations of single- and multi-task learning paradigms, different debate structure models, and varying approaches to text representation and machine learning methods. Our results represent initial benchmarks for this task.

Research questions In this paper, we address the following questions:

RQ1 To what extent do humans agree on the policy preference labelling task? We compare agreement between our annotations with those reported in previous work in both political science (Lacewell and Werner, 2013; Mikhaylov et al., 2008) and natural language processing (Abercrombie et al., 2019). The latter found that agreement was comparable for labels applied to debate motions and the manifestos for which the scheme was originally designed, a finding which we re-examine on this new dataset. **Hypothesis H1:** Policy preference labels are as reliable for debate motions as party-political election manifestos.

RQ2 How well do machine learning classifiers perform on the combined task of policy-focused stance detection? We test a number of approaches against a majority class baseline. These include fine-tuning pre-trained contextual word embeddings, which we compare to a simple bag-of-words model, and a multi-task learning approach designed to take advantage of mutually beneficial information, which we compare to tackling the constituent tasks independently.
Hypothesis H2a: Classification of policy-focused stance will benefit from use of contextual BERT embeddings.
Hypothesis H2b: Classification of policy-focused stance will benefit from concurrent classification of policy preferences and speaker sentiment using a multi-task approach.

2 Background

House of Commons debates As the superior legislative chamber in the UK Parliament, the House of Commons (HoC) draws the attention of the public, the media, and the academic sector, and was therefore chosen as the focus of this study.

Debates in the HoC consist of an opening *motion* (proposal), the content of which usually does not provide clues to the policy that is proposed (see, for example, Figure 1a). We found 75.8 per cent of debate motions in the corpus to contain insufficient information to manually determine a policy preference.

A number of Members of Parliament (MPs) then respond to the motion, when invited to do so by the *Speaker* (the chief presiding officer of the House). An individual MP may make multiple *utterances* during a given debate. Following previous work (Abercrombie and Batista-Navarro, 2020; Salah, 2014; Thomas et al., 2006), we consider a *speech* to be the concatenation of all their utterances in that debate. In many cases, the motion is voted on by MPs in a *division*. As in previous work, we use the record of these votes as labels for sentiment and stance polarity classification.

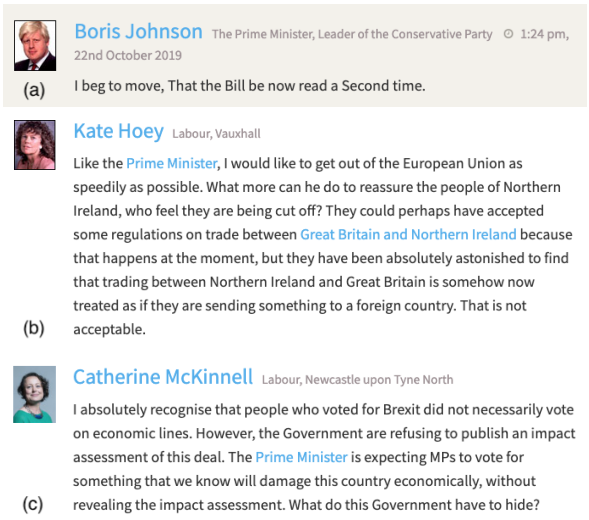


Figure 1: Examples from TheyWorkForYou of *(a)* a debate motion labelled by an annotator with code *110: European Union: Negative*; and two utterances made in response to the motion by speakers who voted *(b) aye* (support) and *(c) no* (oppose).

Policy preferences The concept of policy preferences is widely used in political science (Budge et al., 2001) to categorize the positions of politicians. The Manifesto Project (MARPOR: `https://manifestoproject.wzb.eu`) have developed a set of policy preference codes organised under seven '*domains*'. The current coding scheme comprises 74 policy preference codes, almost all of which are 'positional', encoding a positive or negative position towards a policy issue (Mikhaylov et al., 2008). We use these codes as labels for the policy preferences expressed in the debate

motions. In the example in Figure 1a, the policy preference label applied to this debate by annotators (see §4.1) is *110: European Union: Negative*.

Sentiment and stance detection While use of terminology varies and overlaps in the literature, stance detection can be viewed as a form of sentiment classification. From this perspective, it consists of determining the sentiment polarity of a piece of text towards a predetermined 'given target of interest' (Mohammad et al., 2016). In the case of parliamentary debates, for each example speech, we seek to determine *(1)* the nature of its target—the policy preference under debate—and *(2)* the position or sentiment expressed towards it—*support* or *opposition*. We consider the combined policy preference and speech sentiment labels to represent the speaker's stance on a particular policy. For instance, in the example in Figure 1, the stance of speech extracts *(b)* and *(c)* are *European Union: Negative—support* and *European Union: Negative—oppose*, respectively.

3 Related work

Sentiment classification is one of the the most active areas of research in natural language processing. Within the domain of legislative debates, examples include classification of speeches from the US Congress (Burfoot et al., 2011; Ji and Smith, 2017; Proksch et al., 2019; Thomas et al., 2006), and the UK Parliament (Abercrombie and Batista-Navarro, 2018b, 2020; Bhavan et al., 2019; Salah, 2014; Sawhney et al., 2020). In these works—and in common with ours—speaker sentiment is assumed to be analogous to vote outcome. However, in the task undertaken in these previous works, the nature of the targets—the Bills or motions under debate—is not identified.

The related task of stance detection—in which the target of sentiment *is* (pre-)determined—has been applied to such domains as social media (e.g. Augenstein et al., 2016a,b; Hardalov et al., 2021; Li et al., 2021; Mohammad et al., 2016), online debate forums (e.g. Hardalov et al., 2021; Hasan and Ng, 2013; Somasundaran and Wiebe, 2010; Sridhar et al., 2015), and news articles (Ferreira and Vlachos, 2016; Schiller et al., 2021). For a recent survey, see Küçük and Can (2020).

In most of this work the target is pre-chosen by the user or the system. In the political domain, this has been framed as agreement detection in which two pieces of text are compared (Menini and Tonelli, 2016; Menini et al., 2017), or classification of *support* or *attack* towards pre-defined policies (Menini et al., 2018). While Vamvas and Sennrich (2020) carry out stance detection on the positions expressed by Swiss politicians, they do not perform automatic identification of the policies discussed, only conducting binary *in favour/against* classification in a similar vein to the sentiment/position classification work discussed above.

More similarly to this work, Bar-Haim et al. (2017) used a supervised approach to identify both the stances of extracts from Wikipedia articles and the targets of those stances from a closed list of 'controversial topics'. However, this labelling scheme does not cover the policy positions proposed in parliament.

A common framework for stance detection is the SDQC (Support-Deny-Query-Comment) annotation scheme of Zubiaga et al. (2016). While potentially suitable for our data (*support* and *deny* are equivalent to our *support* and *oppose* labels), application of this framework would require manual annoation of each instance in the dataset with the more fine-grained labels. Instead, we follow the majority of work on legislative debates (e.g. Abercrombie and Batista-Navarro, 2018a; Thomas et al., 2006; Salah, 2014) in taking advantage of pre-existing vote-derived binary labels at the speech level, and thus only requiring the addition of policy preference labels for each debate.

In most of the reviewed work, stance targets are explicitly selected by the authors of the task (e.g. *Donald Trump* (Augenstein et al., 2016a,b), *Richard Nixon* and *John F. Kennedy* (Menini et al., 2018), or *atheism* (Mohammad et al., 2016)). Unlike these, we frame target selection as a multiclass topic classification problem, making use of an existing schema validated by political scientists.

Document classification is an active area of research for tasks such as identification of news and Wikipedia categories (Zhang et al., 2015). For classification of HoC debates, Abercrombie and Batista-Navarro (2018b) used 'policy' labels crowdsourced by the parliamentary monitoring website `https://www.publicwhip.org.uk/` but found this framework limited as it could not be easily scaled up from the small existing labelled dataset. Abercrombie et al. (2019) created a manually annotated dataset of policy preferences in debate motions, and achieved promising results in classifying debate motions according to the MARPOR coding scheme. However, this corpus is unsuitable for our purposes as: *(1)* it does not include speeches made in response to the motions; and *(2)* the motions in this dataset are all *substantive*—that is, they 'express an opinion about something' (Rogers and Walters, 2015), and tend to be of a highly partisan nature, leading to debates in which the stance of MPs can be trivially predicted from their party affiliations. For this study, we seek a mixture of motion types, more represenative of the Hansard record as a whole. Additionally, while they classified debate motions with policy preference labels using textual features derived from the motions themselves, many of the motions in Hansard—and in the corpus used in this study—contain little in the

way of informative textual content (Figure 1a is a typical example). Rather than the motions, we therefore rely on features derived from the response speeches, which we use as input for the classification of both motions and speeches.

Multi-task learning approaches have been taken to many tasks, including part-of-speech tagging, chunking, and named entity recognition (Collobert and Weston, 2008). While such approaches have been applied to sentiment classification of customer reviews (Yu and Jiang, 2016), we are not aware of any uses of multi-task learning in the legislative debate domain. The most common approach to multi-task learning—which we compare with the single task paradigm—is that of hard parameter sharing, first proposed by Caruana (1993).

4 Data

ParlVote (Abercrombie and Batista-Navarro, 2020) is a large corpus (34,010 examples) of HoC debate speeches made between from 1997 and 2019. Each example speech consists of the concatenated utterances of an individual speaker in a given debate, and is presented with the debate motion to which it responds, as well as the vote of the speaker (in support or opposition to the motion), and metadata associated with the debate and the speakers. We adapted this corpus to include an additional, manually annotated policy preference label for each example. As capitalization can be informative in this domain (for example in the terms of address '*Friend*', '*Lady*', '*Gentleman*'), we did not lowercase the text.

4.1 Annotation

We adapted the ParlVote annotation guidelines to include the new codes used in the updated MARPOR *Coding Scheme version 5* (Werner et al., 2015). We make our guidelines available at `https://tinyurl.com/y5twunrm`.

The first author of this paper annotated each debate motion following these guidelines. Included in the guidelines were instructions to code examples featuring the following types of motions with the label *000: No meaningful category applies*:

- *Business of the House* motions, *Programme* motions, other timetabling and procedural motions, and motions to sit in private. Although MPs may use such motions politically, on the face of it they are concerned simply with the running of Parliament, rather than policy.
- Debates with divisions that are not on the motion in question. In many cases the division held at the end of the debate is held on some other point that has been brought up during the debate, such as an amendment introduced by the Speaker.
- Motions that appear to fit several codes, such as *Finance* Bills, *Local Finance* Bills, and Bills concerning the budgets of e.g., Police forces. Within the area of budgetary Bills is the exception of motions debates concerning approval of European Union (EU) Finance Bills, which tend to be positive or negative about the EU.
- Motions concerning constituency boundary changes.

We excluded all examples given this label from the dataset used for the experiments reported below as they cover a wide range of topics and/or do not fit into any of the Manifesto Project codes. While 56 of the policy preference codes were used as labels by the annotators, we also excluded all examples with policy preference codes that occur fewer than 100 times in the dataset, leaving 34 codes used in the classification experiments. This left 23,181 example speeches given by 1,321 unique MPs given in response to 1,215 different debates. Each example has a manually annotated policy preference label and a vote-derived speech stance polarity label. Of these, 305.1: *Political Authority: Party Competence* is the most common, with 4,926 labelled examples (see Appendix A).

Each instance in the corpus also retains it's *support/oppose* label from the original ParlVote corpus, which we use to label the stance taken in each speech towards the policy under debate.

4.2 Inter-annotator agreement

In order to validate the new motion policy preference labels, we recruited a second annotator to label a randomly selected subsection of the corpus. After annotation, comparison, and discussion of some initial training examples, she labelled 108 motions (8.9% of the total). On this subset, we calculated a Cohen's *kappa* agreement score of 0.38, which can be interpreted as representing '*fair*' (Landis and Koch, 1977) or '*poor*' (Fleiss et al., 1981) agreement. This is comparable to other studies of annotation using the Manifesto Project codes (Lacewell and Werner, 2013; Mikhaylov et al., 2008), and similar to agreement on election manifestos for which the labelling scheme was originally designed (Abercrombie et al., 2019). The level of agreement highlights that this is a non-trivial task on which agreement between different human annotators is difficult to achieve. Despite this issue of annotation reproducibility, these labels are considered to be valid by political scientists—as evidenced by Volkens et al. (2015), who found 230 articles that use this annotated data in the

eight journals they examined. With comparable inter-annotator agreement, we consider them to be the best available labelling scheme for our task.

We make the adapted dataset, *ParlVote+*, available for the research community at: `https://tinyurl.com/y22rrta7`.[1] There, we also provide a full data statement, following the guidelines of Bender and Friedman (2018).

5 Method

We investigate approaches to determining, for each example in the dataset, *(a)* the policy preference expressed in the debate motion, and *(b)* the sentiment (position) expressed in the speech towards that motion: *support* (positive) or *oppose* (negative).

We compare the performance of systems comprised of combinations of the following:

- Learning paradigms (see Figure 2):
  - Single tasks: inputs are processed separately for the two tasks, as in previous work.
  - Multi-task learning: we use a 'hard parameter sharing' framework (Ruder, 2017), in which the network shares inputs and parameters in one hidden layer and trains two further task-specific layers separately.
- Debate models:
  - Motion-independent: all examples are trained and evaluated together.
  - Motion-dependent: Abercrombie and Batista-Navarro (2018a) showed that Government-proposed motions tend to be positive and those tabled by opposing parties negative, and that this could be used as a proxy for the polarity of the motions. We classify examples from debates initiated by members of the governing and opposition parties separately.
- Text representations:
  - Bag-of-words (BOW): we used term frequency-inverse document frequency (tf-idf) scores of terms in the dataset to select unigram features (as previous work suggests that the addition of higher *n*-gram features does not improve perfomance in this domain (Abercrombie and Batista-Navarro, 2018a)). Aside from not lowercasing the text, we used the default settings from scikit-learn to tokenize and extract ti-idf features from the texts.[2]
  - Contextual word embeddings: we fine-tuned BERT embeddings (Devlin et al., 2019) on our classification tasks. Systems using this approach have achieved state-of-the-art performances, and have been applied to the two tasks of interest in this domain (Abercrombie et al., 2019; Abercrombie and Batista-Navarro, 2020). As we included uppercase characters in the input, we used the *large, cased* version, available at `https://tfhub.dev/google/bert_cased_L-12_H-768_A-12/`. We use Google's BERT tokenizer,[3] and pad the texts to the maximum input of 512 tokens, then fine-tune the top 3 layers of the BERT model. The (fine-tuned) final layer of BERT embeddings is then used as input to one of the following neural classifiers.
- Machine learning classification algorithms. We used neural networks of two hidden layers, with the second of these separated into two task-specific layers in the multi-task learning setting (see Figure 2). We used Adam optimization with a learning rate of $1 * 10^{-5}$, a batch size of 32 and, with the BOW input only, a dropout rate of 0.5 for each layer.[4] For binary (speech sentiment) and multiclass (motion policy preference), we used sigmoid and softmax activation layers,

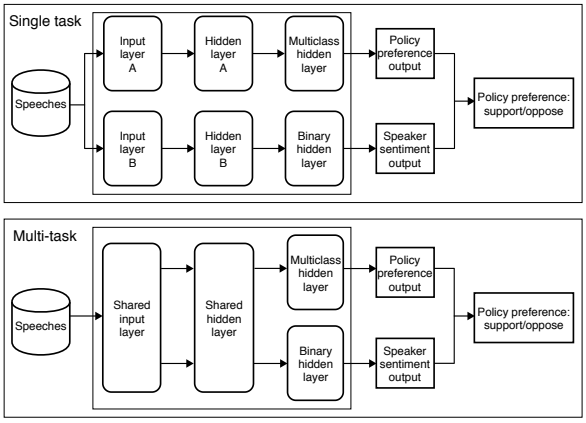


Figure 2: Single and multi-task learning paradigms.

[1] Note this URL links to an anonymised Google Drive folder. Link to a permanent data repository will be provided on acceptance.

[2] `https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html`

[3] `https://github.com/google-research/bert/blob/master/tokenization.py`

[4] Opitimzation experiments showed that dropout negatively influenced the performance using BERT (see Appendix B).

Learning paradigm	Text representation	Machine learning method	Policy pref.		Sentiment		Policy-focused stance			
			Ind.	Dep.	Ind.	Dep.	Mean		Absolute	
—	—	Majority class	1.1	1.1	35.8	35.8	18.5	18.5	0.3	0.3
Single-task	BOW	MLP	58.0	**64.1**	61.2	70.8	59.6	**67.4**	33.3	**45.2**
		CNN	53.1	59.5	61.5	70.1	57.3	64.8	29.9	40.8
	BERT	MLP	50.4	57.2	61.1	67.6	55.8	62.4	28.7	36.4
		CNN	43.0	52.5	64.3	71.7	53.7	62.1	25.2	35.6
Multi-task	BOW	MLP	56.0	52.7	63.9	**74.3**	60.0	63.5	34.1	38.2
		CNN	38.2	38.5	58.5	68.8	48.4	53.7	19.9	21.8
	BERT	MLP	50.9	43.7	60.1	72.8	55.5	58.2	27.9	29.1
		CNN	44.4	41.1	59.4	70.6	51.9	55.8	23.9	25.4

Table 1: Macro-averaged F1 scores for classification of *policy preference* (multiclass), *speech sentiment* (binary), and *policy-focused stance* using motion-independent (*Ind.*) and motion-dependent (*Dep.*) debate models. Stance scores are reported as both the mean of the policy preference and sentiment scores and the absolute F1 score. The highest F1 scores for each task are highlighted in bold text.

respectively. We used early stopping and tested on the model that performed best on the validation set. Hyperparameters were chosen based on optimisation experiments, the results of which are presented in Appendix B.

We compared the following classes of network:

- Multi-layer perceptron (MLP): we used a network with hidden layers of 512 nodes and ReLU activation.
- Convolutional neural network (CNN): a network of one-dimensional convolutional layers with 512 filters, convolution windows spanning three tokens, and max pooling.

We used a randomly sampled 80/10/10 split of the data. The experiments can be reproduced using our python notebook, which we make available with all code and data at `https://tinyurl.com/y62jrkyt`.

6 Results

We evaluated the systems described above against the majority class for each task. Slight differences in these baseline scores in the motion-dependent and independent settings arise from variations in the class distributions in the test sets in these settings. Due to the class imbalances in the dataset, we report the macro-weighted F1 score as the evaluation metric.

6.1 Overall results

Results are presented in Table 1. Here, *policy-focused stance* represents the *sentiment polarity* of speakers towards the *policy preference* under debate. We report two measures of this for each system configuration: *(1)* the mean of the F1 scores for policy preference identification and sentiment classification, and *(2)* the absolute F1 where only examples for which both predicted labels match the true class labels are considered to be correct.

Most of the tested system configurations outperformed the naive baselines. In most cases, the motion dependent models performed better than those that did not take into account this aspect of debate structure. Overall, contrary to our hypotheses, neither BERT nor the multi-task learning paradigm improved performance over the BOW and single-task set-ups. BERT-based systems tended to perform poorly on policy preference identification in the motion-dependent setting, perhaps due to the low number of examples per class combined the with loss of information due to BERT's maximum sequence length. The MLP classifier performed better than the CNN in nearly all scenarios. The highest overall F1 score for the combined tasks (67.4 mean, 45.2 absolute) was obtained by using single task learning with BOW and MLP in the motion-dependent setting. It is notable that the policy preference detection scores (using BOW) are comparable to those obtained by Abercrombie et al. (2019), despite using completely different input texts, having no access to the content of the motions themselves.

6.2 Results using shorter input speeches

The lower, poorer performance of BERT text representations in all settings is perhaps due to its the 512 token sequence input limit. With the mean number of tokens per speech in the ParlVote corpus over 700, in many cases, much potentially important information cannot be included when using this framework. Bearing this in mind, in order to test the potential of BERT for this task, we also ran the single task MLP classifier on a subset of the data consisting solely of the 13,162 speeches in the dataset that consist of 512 tokens or fewer (caluclated using the scikit-learn tokenizer). Results of these experiments are shown in Table 2.

F1 scores here are lower than when using the full

Text representation	Policy preference		Speech sentiment		Policy-focused stance			
					Mean		Absolute	
	Ind.	Dep.	Ind.	Dep.	Ind.	Dep.	Ind.	Dep.
Majority class	0.1	0.1	36.0	36.0	18.1	18.1	0.3	0.3
BOW	32.6	40.9	**56.3**	58.3	**44.5**	49.6	17.7	19.3
BERT	**34.9**	**45.0**	51.0	**62.8**	43.0	**53.9**	**18.5**	**24.8**

Table 2: Macro-averaged F1 scores for classification of policy preference (multiclass), speech sentiment (binary) and policy-focused stance (mean of these scores) using BOW and BERT-based text representations in the *single-task–MLP* classification setting on shorter speeches of 512 tokens or fewer.

Code	Policy pref.	Sentiment	Stance (mean)	Code	Policy pref.	Sentiment	Stance (mean)
104	83.8	68.4	76.1	*411*	**84.6**	**81.2**	**82.9**
105	57.1	47.5	52.3	*413*	45.5	72.7	59.1
106	76.2	61.1	68.7	*501*	65.0	46.4	55.7
108	67.5	58.6	63.1	*503*	46.7	69.3	58.0
110	65.9	54.9	60.4	*504*	65.4	75.0	70.2
201.2	56.9	55.3	56.1	*505*	78.2	67.3	72.8
202.4	76.9	76.4	76.7	*506*	50.0	74.7	62.4
203	31.6	57.8	44.7	*507*	56.1	69.3	62.7
204	69.8	55.2	62.5	*601.2*	36.4	59.0	47.7
301	54.5	67.0	60.8	*602.2*	36.4	47.6	42.0
302	41.0	54.5	47.8	*603*	52.8	60.0	56.4
304	52.6	47.4	50.0	*604*	69.8	53.7	61.8
305.1	83.5	74.6	79.1	*605.1*	79.4	66.4	72.9
305.2	33.3	59.0	46.2	*605.2*	60.8	64.7	62.8
401	51.8	68.3	60.1	*701*	48.5	71.8	60.2
402	44.7	64.4	54.6	*702*	47.1	71.7	59.4
403	61.1	62.0	61.6	*706*	42.1	78.9	60.1

Table 3: F1 scores for policy preference, sentiment, and (mean) policy-focused stance by policy preference code. Highest scores for each task are bold, contrastive pairs of policy preference codes in grey boxes.

dataset due to the smaller size of the training set. However, the fact that under these conditions use of BERT outperforms BOW, shows the importance of providing BERT with the full speech, and indicates that where this is possible fine-tuning on BERT should lead to improved performance over the BOW model.

6.3 Results by policy preference class

Examining the performance of one of the best performing system configurations—the *single-task–BOW–MLP–motion-dependent* system—for each (*true*) policy preference label (Table 3), there are a wide variety of scores for each task.

Each policy preference class received between four and 21 predicted labels in the classifier output (μ = 10.4). Labels with contrastive pairs did not necessarily seem to be more difficult to predict than individual class labels, with, for example *104: Military: Positive* obtaining one of the highest F1 scores for policy preference detection. Similarly, code *411: Technology and Infrastructure: Positive* is in the *Economy domain*, which contains a number of fairly similar codes. However, this code concerns a well defined topic, and has no directly contrastive partner class, and obtained the highest scores overall. This suggests that the model can struggle to differentiate between the closely related, but opposing policy preference classes.

264 examples (22.1% of errors) were classified incorrectly for both policy preference and stance, 520 (43.6%) for policy preference only, and 410 (34.3%) for stance only. Figure 3 shows the predicted policy preference labels with respect to the true labels assigned by the annotators. Where mis-classifications occur, the classifier does not tend to prefer closely related labels, with more than double the number of out-of-*domain* (69.9%) to in-*domain* (31.1%) mis-classifcations. This suggests considerable overlap of language use in policy *domains* such as *4: Economy* and *5: Welfare and Quality of Life*, where issues relating to both may frequently be discussed in the same debates, and on which the annotators frequently disagreed.

6.4 System output analysis

To gain an understanding of the challenges involved in improving classification performance on these tasks, we examined in closer detail the output of the *single-*

	All	+	-	Gov.	Opp.	Own	Other	Gov.+	Gov.-	Opp.+	Opp.-
Max	0.44	0.44	0.28	0.25	0.44	0.38	0.44	0.25	0.23	0.44	0.38
Mean	-0.01	-0.01	-0.02	-0.02	-0.01	-0.01	-0.02	-0.02	-0.02	-0.01	-0.01
Min	-0.38	-0.38	-0.38	-0.38	-0.38	-0.38	-0.38	-0.38	-0.31	-0.38	-0.31

Own +	Own -	Oth. +	Oth. -	Gov. own+	Gov. own-	Gov oth+	Gov- oth-	Opp. own+	Opp. own-	Opp. oth+	Opp. oth-
0.25	0.38	0.25	0.25	0.25	0.21	0.19	0.25	0.25	0.38	0.44	0.28
-0.01	-0.02	-0.01	-0.02	-0.02	0.01	-0.02	-0.01	-0.01	-0.01	-0.01	-0.01
-0.38	-0.31	-0.38	-0.31	-0.31	-0.381	-0.31	-0.38	-0.38	-0.31	-0.38	-0.31

Table 4: Mean sentiment scores for all speeches, supportive (+)/oppositional (-) speeches, replies to *Government*/*opposition* party motions, responses to own/other party motions, and all combinations of these three factors.

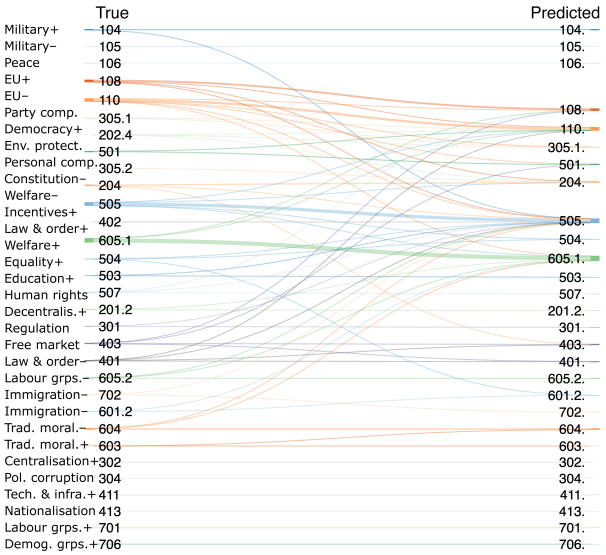


Figure 3: True policy preference labels and the labels predicted by the classifier.

task–BOW–MLP–motion-dependent system.

6.4.1 Features of speech polarity

In these experiments, we found that performance was improved by modelling debate structure in the motion-dependent setting. This supports the findings of Abercrombie and Batista-Navarro (2018a), who observed that the textual features that discriminated between supportive and oppositional speeches were not typically positive or negative when used in other domains.

To investigate how sentiment is manifested in this domain, we first calculated the general-domain sentiment scores of the tokens in each speech example in the test set on a scale of $[-1, 1]$ by looking up the terms in the sentiment lexicon SentiWordNet 3.0 (Baccianella et al., 2010). These scores are shown in Table 4.

The mean sentiment of speeches overall is very slightly negative (-0.01), according to the lexicon. Overall however, there is little difference between supportive and oppositional speeches in the polarity of language used. This is also the case for speeches given in different scenarios, such as in response to *Government*/*opposition* motions, by speakers addressing motions proposed by members with their own or with different party affiliations, or any combinations of these factors. This demonstrates once again that terms used in parliamentary debate speeches do not usually express the same sentiments that they may be expecteed to do in general usage.

To examine which terms in the speeches *do* indicate sentiment, we obtained the permutation importance scores of each unigram in the input vocabulary. That is, for feature j in the feature set N, we calculated the permuation feature importance as the difference between performance (in this case, the F1 score) using the original datset D and a corrupted version $\tilde{D}$, in which j has been randomly shuffled (Breiman, 2001). We consider features with higher scores to be more important to the model. A sample of the most important

Motion-independent		Motion-dependent			
All		Government		Opposition	
Labour	+0.13	*approach*	+0.17	*Minister*	0.00
Gentleman	+0.13	*average*	0.00	*Opposition*	−0.07
shadow	−0.09	*costs*	−0.03	*Prime*	+0.09
Prime	+0.09	*police*	+0.13	*welcome*	+0.19
party	0.00	*contrast*	0.00	*shadow*	−0.09
cuts	+0.01	*registration*	0.00	*Is*	+0.02
Lady	0.00	*officers*	0.00	*continue*	0.00
situation	−0.08	*proposals*	0.00	*best*	+0.38
threat	−0.28	*hit*	−0.03	*look*	0.00
outside	0.00	*tier*	+0.06	*Members*	0.00
pay	+0.06	*fees*	+0.19	*Secretary*	0.00
Lords	0.00	*chance*	+0.08	*ensure*	0.00
crisis	−0.06	*labour*	+0.13	*way*	+0.01
Government	0.00	*constituency*	0.00	*suggestion*	−0.05
constituents	0.00	*dealt*	0.00	*public*	−0.04
wants	−0.06	*running*	0.00	*motion*	0.00
important	+0.08	*data*	0.00	*Clearly*	+0.19
careful	+0.19	*willingness*	+0.13	*support*	+0.09
week	0.00	*tackling*	0.00	*worse*	−0.29
stop	−0.02	*strategy*	+0.06	*said*	0.00

Table 5: Top 20 discriminating features for the motion-independent setting (*all* speeches), and, in the motion-dependent setting, responses to *Government*- and *opposition*-proposed motions, together with their mean SentiWordNet scores.

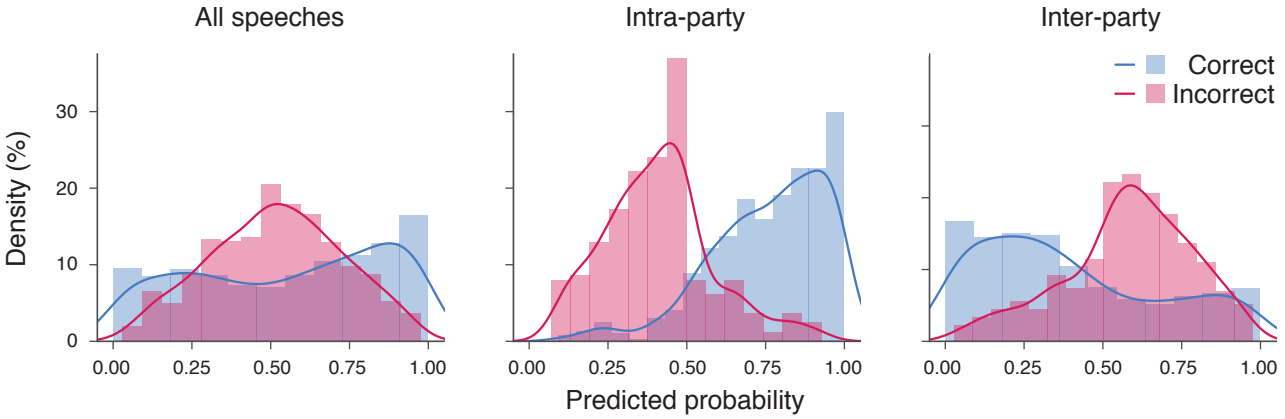


Figure 4: Distribution frequencies (histograms and density curves) of the correct and incorrect predicted probabilities of sentiment labels being positive for three categories of speech: those by *all* speakers, and *intra*- and *inter-party* responses.

features (the top 20) in each setting according to this metric is shown in Table 5

Comparing (the lemmas of) these terms with their SentiWordNet scores (means over all word senses), it seems that the features that are indicative of support or opposition are not those that would typically be used for subjective expression in general English usage. Rather, many are parliamentary terms, such as forms of address, and other proper nouns. This is particularly true for speeches addressing opposition-proposed motions.

6.4.2 Party affiliations

As MPs usually vote along party lines, it would be possible to achieve good sentiment classification results by setting a classifier to make predictions on that simple basis alone. On the other hand, we also know that MPs are more free to 'rebel' against their parties in their speeches than in their voting behaviour (Proksch and Slapin, 2015). To investigate how this effects sentiment polarity classification, we compared the performance of *rebel* MPs—those voting against a motion proposed by their own party or in support of one proposed by an-

	Stance ✓	PP ✓, sent. ✗	Sent. ✓, PP ✗	Stance ✗
n examples	1120	404	492	303
Max. tokens	20730	6505	6484	4742
Mean tokens	876.9	916.5	761.4	867.6
Min. tokens	2	2	2	2
Std. deviation	1213.9	953.2	1115.6	4742
< 50 tokens	103	43	61	35
>= 50 tokens	1017	361	431	268

Table 6: Number of speeches by token counts and prediction outcome (✓= correct and ✗= incorrect).

other party—and *loyal* MPs. This produced F1 scores of 77% and 66% respectively. The lower performance on loyal voters may suggest that, on occasion, speakers may use language that goes some way towards supporting the position of their opponents, while ultimately voting with their parties, and that these cases may be harder to detect than outright rebellions.

The frequency distribution plots in Figure 4 present a closer look at this. They show the predicted probabilites of examples being assigned to the positive class. We compare the probability distributions for correctly and incorrectly predicted testset examples. These densities are shown in three settings: *all* predictions, *intra-party* speeches (made in response to motions proposed by an MP with the same party affiliation), and *inter-party* responses (replies to a member of another party).

There are a number of clear patterns in the distributions. Overall, the system tends to make more confident predictions for examples that it predicts correctly (that is, it outputs probabilities towards 0.0 for negative and 1.0 for positive examples), and is less confident about examples that it predicts incorrectly (closer to 0.5), as might be expected. In the intra-party setting, the model outputs high probablities that it assigns to the positive class (correctly, more often than not). Meanwhile, negative predictions (usually incorrect) are made with probabilites that tend towards 0.5 (that is, with low certainty). For inter-party response speeches, this pattern is reversed, albeit not to as dramatic an extent. This may be due to situations in which, for example, multiple opposition parties collaborate against the Government, which introduce some noise into this analysis. Ultimately, the patterns seen here suggest that the language used in the speeches may often say more about the speakers' party affiliations than it does about about the nuances of individual speaker stance.

6.4.3 Input speech length

The length of speeches does not seem to greatly affect classification, with examples that are classified correctly, partially correctly, and completely incorrectly having similar distributions of token numbers (see Table 6).

Some previous work has excluded speeches of fewer than 50 tokens under the assumption that they are unlikely to contain enough information to express sentiment (Abercrombie and Batista-Navarro, 2018a; Salah, 2014). There are 2,941 such speeches in ParlVote, which are fairly balanced between the positive and negative classes (53/47%) and a very similar distibution of policy preference labels as the main dataset. In the experiments, 67.8% of these shorter examples were classified correctly for speech sentiment (compared with 69.5% of examples of any length), and 42.6% of examples < 50 classified correctly on both tasks (48.1% for the whole dataset). With examples of both very short speeches (such as two-word speeches like '*Hear hear*', '*Under Labour*'–both *negative* stance) and the longest speech examples classified correctly, it seems that speech length is not an important factor in performance for the BOW-based systems.

7 Discussion and conclusion

Policy-focused stance detection of parliamentary speeches is a challenging task, which we have framed as combined binary and multiclass classification. For this, we enhanced an existing dataset with an additional set of policy preference labels. While inter-annotator agreement on policy preference labels is modest, it is similar to that reported in previous work on both parliamentary debates and election manifestos. To address the issue of low annotator agreement, and the fact that classifiers frequently misclassify speeches across policy *domains*, future work could take a *perspectivist* approach to annotator disagreement (Basile et al., 2021a,b), and consider reframing the task as a multiclass *and multilabel* problem, in which more than one policy preference code may be valid per speech. Notwithstanding this issue, and despite the large number of classes in the policy prediction task, and the fact that the input features we used were based only on the content of speeches (not the motions or titles, as in previous work (Abercrombie et al., 2019)), we have been able to obtain reasonable results, comfortably beating the majority

class baselines.

Modelling of the structure of parliamentary debates in the form of motion-dependent classification was seen to improve performance on speech sentiment classification in prior work. In this study, we found that it is not only consistently superior for speech sentiment classification, but also improves the identification of policy preferences, the topics under discussion. We have shown that the differences between supportive and opposing speeches do not derive from generally sentiment bearing words, but from the relationships between the speaker, the MP who proposes the motion in question, and the party affiliations of both actors.

The application of multi-task learning did not, in most configurations, improve system performances. However, we used a fairly simple framework in which just one of the network's hidden layers was shared with one further hidden layer per classification task. There is therefore plenty of scope for further experimentation with more complex architectures for this approach.

In these experiments, fine-tuning on BERT embeddings led to considerably worse performances. Considering the widespread successes of this approach, this also warrants further investigation. With recent work suggesting that, for real-world tasks and datasets, pretraining the embeddings on in-domain data may be necessary (Xia et al., 2020), a more domain-specific approach may be desirable.

While other work on sentiment and stance detection in the domain of parliamentary debates has effectively overlooked the targets of those opinions, we have combined approaches to sentiment and topic detection to formulate a task with potential for real-world application. Although there remains much room for improvement in classification performance, we have shown that the task of policy-focused speech stance detection can be feasibly automated, even with simple features and neural architectures. Although we have focussed our annotation effort and analysis on debates from the UK Parliament, the proposed approach is generalisable to other legislatures, or indeed any political debates that feature proposed motions and supporting and opposing documents.

In future work, we will focus on refining the annotation scheme in order to obtain greater labelling consistency and improved classification performance, as well as adapting the methods for the legislative debate domain.

Acknowledgements

The authors would like to thank the anonymous reviewers for their helpful comments and suggestions and Nancy Greig for her diligent work on data annotation.

References

Abercrombie, Gavin and Riza Batista-Navarro. 2018a. 'aye' or 'no'? speech-level sentiment analysis of hansard UK parliamentary debate transcripts. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Abercrombie, Gavin and Riza Batista-Navarro. 2020. ParlVote: A corpus for sentiment analysis of political debates. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 5073–5078, Marseille, France. European Language Resources Association.

Abercrombie, Gavin and Riza Theresa Batista-Navarro. 2018b. Identifying opinion-topics and polarity of parliamentary debate motions. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 280–285, Brussels, Belgium. Association for Computational Linguistics.

Abercrombie, Gavin, Federico Nanni, Riza Batista-Navarro, and Simone Paolo Ponzetto. 2019. Policy preference detection in parliamentary debate motions. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 249–259, Hong Kong, China. Association for Computational Linguistics.

Augenstein, Isabelle, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. 2016a. Stance detection with bidirectional conditional encoding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 876–885, Austin, Texas. Association for Computational Linguistics.

Augenstein, Isabelle, Andreas Vlachos, and Kalina Bontcheva. 2016b. USFD at SemEval-2016 task 6: Any-target stance detection on Twitter with autoencoders. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 389–393, San Diego, California. Association for Computational Linguistics.

Baccianella, Stefano, Andrea Esuli, and Fabrizio Sebastiani. 2010. SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Bar-Haim, Roy, Indrajit Bhattacharya, Francesco Dinuzzo, Amrita Saha, and Noam Slonim. 2017. Stance

classification of context-dependent claims. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 251–261, Valencia, Spain. Association for Computational Linguistics.

Basile, Valerio, Federico Cabitza, Andrea Campagner, and Michael Fell. 2021a. Toward a Perspectivist turn in ground truthing for predictive computing. In *Conference of the Italian Chapter of the Association for Intelligent Systems (ItAIS 2021)*.

Basile, Valerio, Michael Fell, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, Massimo Poesio, and Alexandra Uma. 2021b. We need to consider disagreement in evaluation. In *Proceedings of the 1st Workshop on Benchmarking: Past, Present and Future*, pages 15–21, Online. Association for Computational Linguistics.

Bender, Emily M. and Batya Friedman. 2018. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6:587–604.

Bhavan, Anjali, Rohan Mishra, Pradyumna Prakhar Sinha, Ramit Sawhney, and Rajiv Ratn Shah. 2019. Investigating political herd mentality: A community sentiment based approach. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 281–287, Florence, Italy. Association for Computational Linguistics.

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

Budge, Ian, Hans-Dieter Klingemann, et al. 2001. *Mapping policy preferences: Estimates for parties, electors, and governments, 1945-1998*, volume 1. Oxford University Press.

Burfoot, Clinton, Steven Bird, and Timothy Baldwin. 2011. Collective classification of congressional floor-debate transcripts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1506–1515, Portland, Oregon, USA. Association for Computational Linguistics.

Caruana, Richard A. 1993. Multitask learning: A knowledge-based source of inductive bias. In *Machine Learning Proceedings 1993*, pages 41 – 48. Morgan Kaufmann, San Francisco (CA).

Collobert, Ronan and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, page 160–167, New York, NY, USA. Association for Computing Machinery.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ferreira, William and Andreas Vlachos. 2016. Emergent: a novel data-set for stance classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1163–1168, San Diego, California. Association for Computational Linguistics.

Fleiss, Joseph L, Bruce Levin, and Myunghee Cho Paik. 1981. *Statistical methods for rates and proportions*. John Wiley & sons.

Hardalov, Momchil, Arnav Arora, Preslav Nakov, and Isabelle Augenstein. 2021. Cross-domain label-adaptive stance detection. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9011–9028, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Hasan, Kazi Saidul and Vincent Ng. 2013. Stance classification of ideological debates: Data, models, features, and constraints. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1348–1356, Nagoya, Japan. Asian Federation of Natural Language Processing.

Ji, Yangfeng and Noah A. Smith. 2017. Neural discourse structure for text categorization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 996–1005, Vancouver, Canada. Association for Computational Linguistics.

Küçük, Dilek and Fazli Can. 2020. Stance detection: A survey. *ACM Comput. Surv.*, 53(1).

Lacewell, Onawa P and Annika Werner. 2013. Coder training: key to enhancing reliability and validity. *Mapping Policy Preferences from Texts*, 3:169–194.

Landis, J. Richard and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Li, Yingjie, Tiberiu Sosea, Aditya Sawant, Ajith Jayaraman Nair, Diana Inkpen, and Cornelia Caragea. 2021. P-stance: A large dataset for stance detection in political domain. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2355–2365, Online. Association for Computational Linguistics.

Menini, Stefano, Elena Cabrio, Sara Tonelli, and Serena Villata. 2018. Never retreat, never retract: Argumentation analysis for political speeches. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Menini, Stefano, Federico Nanni, Simone Paolo Ponzetto, and Sara Tonelli. 2017. Topic-based agreement and disagreement in US electoral manifestos. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2938–2944, Copenhagen, Denmark. Association for Computational Linguistics.

Menini, Stefano and Sara Tonelli. 2016. Agreement and disagreement: Comparison of points of view in the political domain. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2461–2470, Osaka, Japan. The COLING 2016 Organizing Committee.

Mikhaylov, Slava, Michael Laver, and Kenneth Benoit. 2008. Coder reliability and misclassification in Comparative Manifesto Project codings. In *the 66th MPSA Annual National Conference*.

Mohammad, Saif, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. A dataset for detecting stance in tweets. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3945–3952, Portorož, Slovenia. European Language Resources Association (ELRA).

Proksch, Sven-Oliver, Will Lowe, Jens Wäckerle, and Stuart Soroka. 2019. Multilingual sentiment analysis: A new approach to measuring conflict in legislative speeches. *Legislative Studies Quarterly*, 44(1):97–131.

Proksch, Sven-Oliver and Jonathan B Slapin. 2015. *The politics of parliamentary debate*. Cambridge University Press.

Rogers, Robert and Rhodri Walters. 2015. *How Parliament works*. Routledge.

Ruder, Sebastian. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*.

Salah, Zaher. 2014. *Machine learning and sentiment analysis approaches for the analysis of Parliamentary debates*. Ph.D. thesis, University of Liverpool.

Sawhney, Ramit, Arnav Wadhwa, Shivam Agarwal, and Rajiv Ratn Shah. 2020. GPolS: A contextual graph-based language model for analyzing parliamentary debates and political cohesion. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4847–4859, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Schiller, Benjamin, Johannes Daxenberger, and Iryna Gurevych. 2021. Stance detection benchmark: How robust is your stance detection? *Künstliche Intelligenz*.

Somasundaran, Swapna and Janyce Wiebe. 2010. Recognizing stances in ideological on-line debates. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 116–124, Los Angeles, CA. Association for Computational Linguistics.

Sridhar, Dhanya, James Foulds, Bert Huang, Lise Getoor, and Marilyn Walker. 2015. Joint models of disagreement and stance in online debate. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 116–125, Beijing, China. Association for Computational Linguistics.

Thomas, Matt, Bo Pang, and Lillian Lee. 2006. Get out the vote: Determining support or opposition from congressional floor-debate transcripts. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 327–335, Sydney, Australia. Association for Computational Linguistics.

Vamvas, Jannis and Rico Sennrich. 2020. X-stance: A multilingual multi-target dataset for stance detection.

Volkens, Andrea, Cristina Ares, Radostina Bratanova, and Lea Kaftan. 2015. Scope, range, and extent of Manifesto Project data usage: A survey of publications in eight high-impact journals. In *Handbook for Data Users and Coders*. WZB.

Werner, Annika, Onawa Lacewell, and Andrea Volkens. 2015. Manifesto coding instructions: 5th fully revised edition.

Xia, Patrick, Shijie Wu, and Benjamin Van Durme. 2020. Which *BERT? A survey organizing contextualized encoders. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7516–7533, Online. Association for Computational Linguistics.

Yu, Jianfei and Jing Jiang. 2016. Learning sentence embeddings with auxiliary tasks for cross-domain sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 236–246, Austin, Texas. Association for Computational Linguistics.

Zhang, Xiang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 649–657. Curran Associates, Inc.

Zubiaga, Arkaitz, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Peter Tolmie. 2016. Analysing how people orient to and spread rumours in social media by looking at conversational threads. *PLOS ONE*, 11(3):1–29.

A The ParlVote+ corpus

Table 7 shows the number of example speeches that are labelled with each of the MARPOR codes.

B Machine learning parameter optimisation results

Results of preliminary experiments to select the optimal size of CNN window, number of layers of BERT to fine-tune, and dropout rate are shown in Tables 8, 9, and 10.

For the main experiments, the results of which are presented in Section 6, we selected the parameters that resulted in highest F1 scores in the majority of settings in these preliminary tests: CNN window size of 3, fine-tuning three layers of BERT, and a dropout rate of 0.5 in the BOW setting, with no dropout when using BERT.

Code	Name	*n*	Code	Name	*n*
000	*No meaningful category*	9524	*407*	*Protectionism: Neg.*	43
101	*Foreign Relationships: Pos.*	48	***411***	***Technology: Pos.***	137
102	*Foreign Relationships: Neg.*	12	***413***	***Nationalisation***	**254**
104	***Military: Pos.***	**398**	*414*	*Economic Orthodoxy*	54
105	***Military: Neg.***	**181**	*416.2*	*Sustainability: Pos.*	13
106	***Peace***	**155**	***501***	***Environ. Protection***	**631**
107	*Internationalism: Positive*	67	*502*	*Culture: Positive*	14
108	***European Union: Pos.***	**1601**	***503***	***Equality: Positive***	**1336**
109	*Internationalism: Neg.*	13	***504***	***Welfare State Expansion***	**1410**
110	***European Union: Neg.***	**1063**	***505***	***Welfare State Limitation***	**976**
201.2	***Human Rights***	**469**	***506***	***Education Expansion***	**269**
202.2	*Democracy—General: Pos.*	3	***507***	***Education Limitation***	**404**
202.3	*Repr. Democracy: Pos.*	1	*601.1*	*National Way of Life: Pos.*	11
202.4	***Direct Democracy: Pos.***	**166**	***601.2***	***Immigration: Neg.***	**198**
203	***Constitutionalism: Pos.***	**144**	***602.2***	***Immigration: Pos.***	**173**
204	***Constitutionalism: Neg.***	**437**	***603***	***Traditional Morality: Pos.***	**326**
301	***Decentralisation: Pos.***	**570**	***604***	***Traditional Morality: Neg.***	**527**
302	***Centralisation: Pos.***	**398**	***605.1***	***Law and Order: Pos.***	**1399**
303	*Govt. and Admin. Efficiency*	59	***605.2***	***Law and Order: Neg.***	**602**
304	***Political Corruption***	**276**	*606.1*	*Civic Mindedness: Pos.*	11
305.1	***Political Auth.: Party***	**4926**	*607.2*	*Multiculturalism: Pos.*	4
305.2	***Political Auth.: Personal***	**312**	*608.2*	*Multiculturalism: Neg.*	14
401	***Free Market Economy***	**1061**	***701***	***Labour Groups: Pos.***	**576**
402	***Incentives: Positive***	**402**	***702***	***Labour Groups: Neg.***	**186**
403	***Market Regulation***	**988**	*703.1*	*Agriculture and Farmers: Neg.*	25
405	*Corporatism/Mixed Economy*	2	*705*	*Middle Class and Prof. Groups*	78
406	*Protectionism: Positive*	40	***706***	*Underprivileged Min. Groups*	**230**

Table 7: Number of examples in the dataset labelled with each MARPOR *policy preference* code used. Codes used as class labels in the classification experiments described in Section 5 are highlighted in bold text.

Window size	Text representation	Learning paradigm	Policy pref.		Sentiment		Policy-focused stance			
			Ind.	Dep.	Ind.	Dep.	Mean		Absolute	
3	BOW	STL	**53.1**	**59.1**	**61.5**	**70.1**	**57.3**	**64.8**	**30.0**	**40.8**
		MTL	**38.2**	**38.5**	**58.5**	**68.8**	**48.4**	**53.7**	**19.9**	**21.8**
	BERT	STL	**43.5**	**51.3**	**64.0**	**70.1**	**53.8**	**60.1**	**26.3**	**33.9**
		MTL	38.3	**42.8**	56.3	71.0	47.3	**56.9**	18.0	**27.9**
4	BOW	STL	32.6	40.4	56.3	58.2	44.5	49.3	17.7	19.3
		MTL	1.0	1.45	36.0	37.6	18.5	19.5	0.3	0.5
	BERT	STL	21.5	30.6	54.7	64.0	38.1	47.3	11.4	16.4
		MTL	36.7	25.4	57.2	71.8	46.9	48.6	18.0	16.2
5	BOW	STL	52.5	41.4	61.0	66.2	56.8	53.8	29.6	23.9
		MTL	0.40	–	36.0	–	18.2	–	0.1	–
	BERT	STL	21.8	26.9	50.9	51.1	36.3	39.0	10.5	12.1
		MTL	**39.4**	29.7	**57.8**	**72.3**	**48.6**	51.0	**21.5**	20.3

Table 8: Macro F1 scores for classification using CNN with windows of three, four, and five tokens.

Fine-tune layers	Learning paradigm	Policy pref.		Sentiment		Policy-focused stance			
		Ind.	Dep.	Ind.	Dep.	Mean		Absolute	
3	STL	**50.4**	**57.2**	**61.2**	67.6	**55.8**	62.4	**28.7**	**36.4**
	MTL	**50.9**	43.7	60.1	**72.8**	**55.5**	**58.2**	**27.9**	**29.1**
6	STL	45.7	53.6	61.1	**73.0**	53.4	**63.3**	24.9	**37.1**
	MTL	36.0	36.2	**64.7**	63.9	50.4	50.1	21.0	21.4
9	STL	41.1	54.0	59.7	70.8	50.4	62.4	22.5	35.7
	MTL	37.7	**45.2**	63.4	60.2	50.5	52.7	22.8	24.7

Table 9: Macro F1 scores for classification using MLP and fine-tuning three, six, and nine of the 12 BERT layers. Highest F1 scores for each learning paradigm are presented in bold, absolute highest scores are underlined.

Dropout rate	Text representation	Learning paradigm	Policy pref.		Sentiment		Policy-focused stance			
			Ind.	Dep.	Ind.	Dep.	Mean		Absolute	
0.5	BOW	STL	**58.0**	**64.1**	**61.2**	**70.8**	**59.6**	**67.4**	**33.3**	**45.2**
		MTL	**56.0**	**52.7**	63.9	**74.2**	**60.0**	**63.4**	**34.1**	38.2
	BERT	STL	47.5	53.2	60.0	**70.6**	53.7	61.9	24.9	35.6
		MTL	41.3	31.3	**62.6**	**74.5**	51.9	52.9	24.7	22.1
0.2	BOW	STL	54.0	60.3	59.3	68.9	56.6	64.6	30.1	42.0
		MTL	53.6	51.1	**64.0**	**74.2**	58.8	62.6	32.4	**38.5**
	BERT	STL	48.2	54.5	57.5	69.0	52.8	61.7	25.4	35.0
		MTL	46.5	39.4	62.4	72.2	54.5	55.8	25.7	24.8
0.0	BOW	STL	50.7	56.7	57.9	68.1	54.3	62.4	28.4	38.5
		MTL	49.9	47.7	63.2	73.7	56.6	60.7	29.3	35.9
	BERT	STL	**50.4**	**57.2**	**61.2**	67.6	**55.8**	**62.4**	**28.7**	**36.4**
		MTL	**50.9**	**43.7**	60.1	72.8	**55.5**	**58.2**	**27.9**	**29.1**

Table 10: Macro F1 scores for classification using MLP and different dropout rates: 0.5, 0.2, 0.0 (no dropout). For each task and setting, highest F1 scores for each combination of text representation and learning paradigm are presented in bold, absolute highest scores are underlined.

Spanish Abstract Meaning Representation: Annotation of a General Corpus

Shira Wein, Georgetown University, Washington, DC, USA sw1158@georgetown.edu
Lucia Donatelli, Saarland University, Germany donatelli@coli.uni-saarland.de
Ethan Ricker, Georgetown University, Washington, DC, USA ear131@georgetown.edu
Calvin Engstrom, Georgetown University, Washington, DC, USA cle41@georgetown.edu
Alex Nelson, Georgetown University, Washington, DC, USA amn106@georgetown.edu
Leonie Harter, Saarland University, Germany leonie-harter@web.de
Nathan Schneider, Georgetown University, Washington, DC, USA nathan.schneider@georgetown.edu

Abstract Abstract Meaning Representation (AMR), originally designed for English, has been adapted to a number of languages to facilitate cross-lingual semantic representation and analysis. We build on previous work and present the first sizable, general annotation project for Spanish AMR. We release a detailed set of annotation guidelines and a corpus of 486 gold-annotated sentences spanning multiple genres from an existing, cross-lingual AMR corpus. Our work constitutes the second largest non-English gold AMR corpus to date. Fine-tuning an AMR-to-Spanish generation model with our annotations results in an absolute BERTScore improvement of 8.8%, demonstrating initial utility of our work.

1 Introduction

Abstract Meaning Representation (AMR) represents the core meaning of a sentence as a directed, rooted graph focused on predicate-argument structure (Banarescu et al., 2013) (figure 1). Nodes correspond to concepts and labels denote relations between concepts. Labels can be core roles functioning as predicates or arguments, or other attributes such as `:location` or `:manner`.

While there are large AMR-annotated corpora available for English, cross-lingual adaptations of AMR are necessary if AMR is to be useful as an interlingua or intermediate representation for cross-lingual tasks (Xue et al., 2014). Recent work has adapted AMR to a variety of languages (§2.1), evaluating cross-lingual efficacy of rolesets, word senses, and how effectively AMR relations capture "who is doing what to whom" in languages other than English.

As AMR aims to abstract away from morphosyntax, its graph structure is closer to logic than a syntactic parse. For English, AMR removes information such as number, definiteness, tense, word class, and word order. Yet, in many languages, morphosyntactic information in languages other than English carries rich, important semantic information beyond the "sugar" AMR intends to avoid. Therefore, it is important when developing non-English AMR annotation schema to both consider consistency with work in other languages (primarily English) as well as effectively reflecting the semantics of the language being annotated as much as possible.

Spanish is one of the most widely spoken languages in the world. There has been one previous proposal for adapting AMR to Spanish: Migueles-Abraira et al. (2018) presented a corpus of 50 representative annotations for a Spanish translation of (*The Little Prince*) (LPP) (§2.2). While Migueles-Abraira et al. (2018) noted that English AMR failed

```
(a) (s / say-01
        :ARG0 (p / prince
            :mod (l / little))
        :ARG1 (h / hurry-01)
            :ARG1 (t / they)
            :degree (g / great))
(b) (d / decir-01
        :ARG0 (p / príncipe
            :mod (p2 / pequeño))
        :ARG1 (a / apresurado
            :domain (t / th-pers-pl-sinnombre)
            :degree (m / muy)))
```

Figure 1: English (a) and Spanish (b) AMRs for the sentence *"They are in a great hurry," said the little prince.* (*"Tienen mucha prisa," dijo el principito.*) in PENMAN/text-based notation. The Spanish annotation from Migueles-Abraira et al. (2018) is adapted to our schema; `th-pers-pl-sinnombre` is an abbreviation of `third-person-plural-sinnombre` (§3.5) in this example AMR.

to adequately capture semantic phenomena in Spanish, they indicated that accurate representation could be accomplished by adding specific roles and constructions. For example, the English and Spanish AMRs in figure 1, which annotate parallel sentences, have two syntactic divergences due to inherent differences between the languages (Wein and Schneider, 2021).

We extend this prior work on Spanish AMR and present the first substantial Spanish AMR corpus of 486 gold-annotated Spanish AMRs (§4). Specifically, we annotate the Spanish sentences from the "Abstract Meaning Representation 2.0 - Four Translations" dataset (Damonte and Cohen, 2020), a corpus from the news domain that has become a popular resource for evaluation of cross-lingual AMR parsers (Blloshmi et al., 2020; Procopio et al., 2021; Cai et al., 2021) and that spans more genres than LPP.

To support the annotation, we develop annotation guidelines that update and complete those previously established for Spanish (§3). As with prior work, we find that AMR's principle of abstracting away from morphosyntax creates challenges for representing meaning in agreement-rich languages such as Spanish; we present solutions that may be extendable to other languages that exhibit similar linguistic phenomena (§3.14). Our work adds to the development of non-English AMR schema and discusses how to balance consistency and compatibility with standard English AMR while capturing pertinent semantic information not explicitly encoded in English. Three annotators were involved (§4); their work is verified with detailed analysis of inter-annotator agreement and disagreement (§5). Our annotations are publicly available on GitHub.[1]

Finally, to underscore the utility of our gold annotations, we conduct an initial evaluation for a cross-lingual generation task (§6). We show that by fine-tuning an AMR-to-Spanish generation model we are able to achieve an 8.8% increase in BERTScore (Zhang et al., 2019) performance.

2 Related Work

2.1 Cross-lingual Adaptations of AMR

Though AMR was originally designed for English (Banarescu et al., 2013), AMR's abstraction away from morphosyntactic variation lends itself to cross-lingual adaptation by capturing shared semantic structure (Li et al., 2016). Cross-lingual adaptations of AMR have been developed and evaluated for Czech (Hajič et al., 2014), Chinese (Xue et al., 2014; Li et al., 2016), Spanish (Migueles-Abraira et al., 2018), Vietnamese (Linh and Nguyen, 2019), Korean (Choe et al., 2020), Portuguese (Sobrevilla Cabezudo and Pardo, 2019; Anchiêta and Pardo, 2018; Inácio et al., 2022), Turkish (Azin and Eryiğit, 2019; Oral et al., 2022), Persian (Takhshid et al., 2022), and Celtic languages (Heinecke and Shimorina, 2022).

Abstraction can also create challenges, such that changes are required to the annotation schema to sufficiently account for language variation and pertinent linguistic phenomena in non-English AMR. For example, a comparison between English and Czech AMRs found that only 29 of 100 AMRs shared identical structure, and that key differences arose in event structure, multi-word expressions, and compound nouns (Xue et al., 2014).

[1] The annotations are available at `https://github.com/shirawein/Spanish-Abstract-Meaning-Representation.git`. The associated sentences are available through the Linguistic Data Consortium.

2.2 Prior Work Adapting AMR to Spanish

Prior work has proposed an initial adaptation of AMR to Spanish (Migueles-Abraira et al., 2018) using English AMR guidelines (Banarescu et al., 2019) as a baseline to pilot annotation for Spanish sentences. Seven key linguistic phenomena were identified as necessary to add to English AMR to capture essential semantic information in Spanish: (1) NP ellipsis, (2) third person possessive pronouns, (3) third person clitic pronouns, (4) varied *se* usage, (5) gender, (6) verbal periphrases/verbal structure and locutions, and (7) double negatives. Guidelines were developed for the first four of these phenomena, and 50 representative sentences of the Spanish translation of *The Little Prince* were annotated. Spanish translations were made to be more literal so that they would be more semantically equivalent to the original translation of the work.

One limitation of the previous approach was the use of English PropBank (Kingsbury and Palmer, 2002; Palmer et al., 2005) for sense annotation instead of AnCora (Taulé et al., 2008) (§4.4), a similar resource developed for Spanish. English PropBank senses do not correspond one-to-one with their Spanish verbs and bias word meanings towards English-based semantics. Migueles-Abraira et al. (2018) chose rolesets from English PropBank instead of AnCora as it provided more coverage of words in the corpus. Spanish words were translated to English, and the sense from the English word was attached to the Spanish word (Migueles-Abraira, 2017).

A second limitation of the previous Spanish AMR annotation was the limited amount of change to the English AMR guidelines to incorporate Spanish linguistic phenomena. Recent work has assessed various differences between Spanish and English annotations of the existing Spanish AMR adaptation, classifying the type and cause of the identified differences (Wein and Schneider, 2021).

3 Aims and Guidelines

Our primary aims with the development of this corpus included the release of a (1) sizable, (2) general-purpose Spanish AMR corpus, which can be useful in the evaluation of cross-lingual AMR parsers, (3) which effectively represents Spanish semantics. We set out to meet these goals by (1) manually annotating 586 AMRs, (2) annotating the Four Translations dataset, often used for evaluation of cross-lingual AMR parsers, and (3) developing guidelines which consider a range of linguistic phenomena. In this section, we discuss the key considerations and linguistic phenomena we prioritize in our approach to Spanish AMR annotation.

3.1 Use of English and Connection to English AMR Guidelines

Our guidelines are developed in reference to the English AMR Guidelines,[2] outlining the differences between our annotation schema of Spanish sentences and the annotation for English AMRs. As has been popularized in other non-English AMR corpora (Linh and Nguyen, 2019; Sobrevilla Cabezudo and Pardo, 2019), we maintain the role labels and canonical entity type list in English. For example, we use `:ARG0`, `:ARG1`, etc., as well as `:domain`, `:time`, etc., and `person`, `government-organization`, `location`, etc.

3.2 Verb Senses

We number verb senses according to the AnCora lexicon,[3] and supplement these with new senses for out-of-vocabulary lexemes and meanings encountered in our data (table 1). Usage examples for these senses are included in the guidelines.

3.3 Modality

The modal verbs *deber* ("must", "should") and *poder* ("might", "could") appear in table 1 in the list of words which appear in AnCora with other senses. Though meanings of *deber* and *poder* do appear in AnCora, we establish additional senses to mark modality. These modals take the same `:ARG1` structure as do their English modal equivalents—`recommend-01` and `possible-01`, respectively. These modals take the verb senses `deber-03` and `poder-04`.

[2] https://github.com/amrisi/amr-guidelines/blob/master/amr.md

[3] http://clic.ub.edu/corpus/en/ancoraverb_es

Verb	AnCora?	S#	English Translation
auditar	no	-01	to audit
disuadir	no	-01	to dissuade
vagar	no	-01	to wander
hervir	no	-01	to boil
desvanecer	yes	-02	to fade
sobrecargar	no	-01	to overload
congestionar	no	-01	to congest (traffic)
incriminar	no	-01	to incriminate
circunvalar	no	-01	to encircle
adular	no	-01	to flatter
salir	yes	-11	to go out (with someone)
entrelazar	no	-01	to interlace
zonificar	no	-01	to zone
embotellar	no	-01	to bottle up
deber	yes	-03	[modal] to recommend
poder	yes	-04	[modal] to be possible

Table 1: Table of verb senses specification for annotation of senses which are not covered by AnCora. The "Ancora?" column indicates whether the verb is included at all in AnCora, for any senses. If the verb appears for other sense, the S# (sense number) increases to the next available label.

3.4 Gender

In Spanish, all nouns have lexical gender (masculine or feminine), which affects agreement. Nouns relating to humans or animals will also be marked with natural/interpretable gender, such as *hermano* ("brother") versus *hermana* ("sister"). Either way, we remove only number information when lemmatizing the word for AMR, so *niños* (whether it means "boys", or "boys and girls") will always be represented with the concept *niño*, and *niñas* ("girls") with *niña*. If any agreeing adjectives appear, the gendered (singular) concept is annotated.

3.5 Pronoun Drop

Spanish belongs to a group of languages that allow pronoun drop (pro-drop), in which certain pronouns can be omitted if they are grammatically or pragmatically inferable from the surrounding linguistic context. Pro-drop in Spanish occurs only with subject pronouns and is permitted only in certain contexts (Española, 2010).[4] Migueles-Abraira et al. (2018) specify a special concept `sinnombre` ("nameless") for implicit references where no antecedent in context is represented in the AMR. We refine this approach to also encode person and number for these implicit entities following the standard format: `first-person-sing-sinnombre`, `first-person-plural-sinnombre`, etc.

For example, in *No sé que quiero* ("I do not know what I want"), there is an implicit subject *yo* ("I") that is reflected in the verbal agreement. We therefore specify `first-person-sing-sinnombre` as the agent. We choose to use `first-person-sing-sinnombre` instead of the reentrant `yo` ("I") as the conditions on the use of overt and dropped pronouns are typically subject to information structure, an important component of sentence meaning.

No sé que quiero. ("I do not know what I want.")

```
(s / saber-01
    :polarity -
    :ARG0 (f / first-person-sing-sinnombre)
    :ARG1 (h / querer-01
        :ARG0 f))
```

If the pronoun is present (e.g. *él, ella, usted,* etc.), the pronoun should be used in place of a `sinnombre` concept.

3.6 Polite Second Person Addressee

Usted ("you") can reflect either a polite usage of second person, or third person. When *usted* is used as a polite second person pronoun, the polite modifier should be added: `:mod-polite +`. This follows the same structure as `:polarity -`.

3.7 Third Person Possessives

We treat third person possessives similarly to the English annotation, using the `sinnombre` concepts discussed above. For example, we annotate *su coche* ("his car") the same way that "his car" is structured.

his car

```
(c / car
    :poss (h / he))
```

[4] Subject drop is viable in Spanish due to inflection of person and number in the verb. Other pro-drop languages permit the elision of pronouns in other positions. Future work can look at the impact of AMR's abstraction away from morphosyntactic information that allows phenomena such as pro-drop, especially in translation and generation tasks.

su coche (“his car”)

```
(c / coche
    :poss (e / third-person-sing-sinnombre))
```

The possessive pronoun *su* is ambiguous (“his”/“hers”/“its”), and could be annotated as `third-person-sing-sinnombre` (in the case of “his”), `second-person-sing-sinnombre` (as in “yours”), or `third-person-plural-sinnombre` (for “theirs”). These labels are only required when the use of *su* as a possessive pronoun is ambiguous. For example, in the case of *Sofía me mostró su auto* (“Sofía showed me her car”), *su* very likely refers to Sofia’s. However, in *Sofía copió su tarea* (“Sofía copied their homework”), this likely means that Sofia copied someone else’s homework; *su* would refer to some unnamed person, and would thus require the use of `third-person-sing-sinnombre`. Because *su* covers all third person possessives, this distinction requires some interpretation by the annotator based on context and meaning.

3.8 Third Person Clitic Pronouns

Clitics are treated as separate concepts, following (Migueles-Abraira et al., 2018). For example, *mandarlo* (“send it”) has a root of *mandar* (“send”) and an ARG1 of the item being sent: *lo* (“it”).

```
(m / mandar-01
    :ARG1 (l / lo))
```

3.9 Se Usage

Se has many uses in Spanish, including: (1) as a reflexive pronoun, (2) to denote the passive voice, (3) as a substitute for the indirect pronoun *le/les*, and (4) as an impersonal pronoun.

Se as a Reflexive Pronoun. Reflexives are represented via reentrancies as in English AMR. Two examples include the use of *se* in *ellos se perjudican* (“they are harmed”) and in *Pablo se ve* (“Pablo sees himself”).

Ellos se perjudican. (“They harm themselves.”)

```
(p / perjudicar-01
    :ARG0 (e / ellos)
    :ARG1 e)
```

Pablo se ve. (“Pablo sees himself.”)

```
(v / ver-01
    :ARG0 (p / person
        :name (n / name
            :op1 "Pablo"))
    :ARG1 p)
```

Se as a Passive Marker. When *se* reflects a passive voice for an omitted concept, we use the `:ARG0` role label with *se*.

Se venden casas rurales. (“Rural houses for sale.”)

```
(v / vender-01
    :ARG0 (s / se)
    :ARG1 (c / casa
        :mod (r / rural)))
```

Se as an Impersonal Pronoun. *Se* used to mean “one” is annotated with the concept `se-impersonal`.

No se debe beber. (“One should not drink.”)

```
(d / deber-03
    :polarity -
    :ARG0 (b / beber-01)
    :ARG1 (s / se-impersonal))
```

3.10 Double Negation

In Spanish, negation can be indicated by either single or double negatives, with double negatives sometimes providing emphasis. We annotate both single and double negation with the use of one polarity marker.

No hay ninguna persona. (“There is nobody.”)

```
(h / haber-01
    :polarity -
    :ARG0 (p / persona))
```

3.11 Suffixes

Derivational suffixes such as diminutives should be represented as modifier concepts. For example, *poquito* (“very little”) would be annotated with *poco* (“little”) being modified by *muy* (“very”).

```
(p / poco
    :mod (m / muy))
```

Another example would be *hombrecito* ("little man"), for which would *hombre* ("man") receive the diminutive modifier of *pequeño* ("little").

```
(h / hombre
    :mod (p / pequeño))
```

3.12 Words that Change Meaning When Singular Or Plural

In Spanish AMR as in English AMR, we annotate the concept as the singular of the entity even if it is plural. However, rarely in Spanish a word changes meaning if it is plural instead of singular. In this case we use the plural form of the word, such as *deber* (duty) versus *deberes* (homework), or *resto* (remainder) versus *restos* (human remains or rubbish). Additionally, we distinguish *algún* from *algunos*, for the case in which *algún* means "any" and *algunos* means "some." Similarly, we distinguish *otros* ("others") as a plural noun to mean a distinct group of "others," and preserve the plural *otros* instead of making it singular as *otro* ("other").

3.13 Comparison with Previous Work

The most notable difference between our approach and that of Migueles-Abraira et al. (2018) is that theirs uses Spanish labels while ours uses English labels. Additional differences are largely due to our choice to break down the unnamed category of dropped entities into subcategories based on the type of noun phrase or pronoun. For NP ellipses (§3.5) and third person possessives (§3.7), we use the 6 tags outlined, which specify person and number. Migueles-Abraira et al. (2018) uses a standardized `ente` ("being") concept with `sinnombre` ("nameless") argument for NP ellipses and a `sinespecificar` ("unspecified") argument for third person possessives. In comparison to our annotation in §3.7 for *su coche* ("his car"), the annotation in the corpus from Migueles-Abraira et al. (2018) separates entities (`ente`) and the possessive pronoun itself. Notably, this annotation focuses more on the morphosyntax than semantics:

```
(c / coche
    :posee (e / ente
        :sinespecificar (s / su)))
```

Our approach as well as that of Migueles-Abraira et al. (2018) represents clitics as if they were separated from the stem. We also both approach *se* as a reflexive pronoun in the same way via reentrancy. However, the approach of previous work omits *se* when it is used in the impersonal or passive voice, which we include via the `se-impersonal` concept and `ARG0` label, respectively (§3.9). We also address the issues of *se* as a substitute for *le* or *les* (§3.9), modality (§3.3), gender (§3.3), polite use of *usted* ("you") (§3.6), double negation (§3.10), diminutive and augmentative suffixes (§3.11), meaning change in the singular versus plural (§3.12), and commas/decimals.

3.14 Limitations

Adapting standard English AMR to Spanish involves striking a balance between faithfully capturing the semantics of the Spanish sentence on the one hand, and mirroring the English annotation schema on the other. Here we discuss a few challenges.

Gender and Number Marking. The construction of Spanish interpretable/natural gender and its relationship to morphosyntax are open questions (Donatelli, 2019). In our annotation schema, we opted for simplicity, choosing not to explicitly annotate gender, but to leave any gender-bearing morphology as is in the concept. Migueles-Abraira (2017) encodes gender explicitly by converting all nouns to their masculine form, and adding a `:masc` or `:fem` role label.

Like in English AMR, number inflection is removed unless that would alter the meaning of the stem (§3.12). The possibility of encoding number and gender more explicitly is left to future work.

Idiomatic Expressions. Idiomatic expressions are difficult to annotate with AMR. As is the case for English, Spanish has numerous idiomatic expressions, phrases that have a meaning different to that of individual words in the phrase. Idiomatic expressions are annotated on a case-by-case basis. In the corpus, the majority of idiomatic expressions are either condensed into one concept (*por supuesto*, "of course," becomes `por-supuesto`), or we must use a similar, pre-existing verb to convey the expression's meaning, such as *tener prisa* ("to be in a rush").

Limitations with AnCora. AnCora's predicate lexicon only includes verbs, unlike English Prop-

Bank (Palmer et al., 2005), which has been extended beyond verbs to include noun, adjective, and complex predicates (Bonial et al., 2014). AnCora notably lacks adjective frames and numerous idiomatic/phrasal verbs. This posed a challenge when annotating many adjectives and (often more colloquial) verb phrases. When handling idiomatic verb usage, it is easy (but problematic) for annotators to default to using the structure of the equivalent English idiomatic structure, and substitute Spanish tokens into the English structure. Some AnCora rolesets were missing important core roles. Expanding AnCora or other Spanish propbank efforts would enhance any AMR annotations relying on it.

Mood. Spanish exhibits three grammatical moods: indicative, imperative, and subjunctive. English AMR assumes all sentences to be in indicative mood unless otherwise marked. There are two categories for additional moods: imperatives are marked with `:mode imperative` and expressive utterances with `:mode expressive`. As this is a very rudimentary treatment of the semantics of mood, we choose not to adapt it for Spanish AMR. Future work will look at how to integrate the subjunctive mood into Spanish AMR at both the verbal and sentential levels.

4 Annotation Methodology

4.1 Dataset

We perform annotations on the "AMR 2.0 - Four Translations" dataset, which is released through the Linguistic Data Consortium (Damonte and Cohen, 2020) and has become a popular evaluation tool for cross-lingual AMR parsers (Blloshmi et al., 2020; Procopio et al., 2021; Cai et al., 2021). This dataset contains gold AMRs for English test split sentences from the AMR Annotation Release 2.0 (Knight et al., 2017) alongside translations of those sentences into Italian, Spanish, German, and Mandarin Chinese. The sentences originate mostly from news sources, including broadcast conversations, newswire and web text—genres broader than but complementary to the LPP corpus often used for AMR annotation. The corpus contains 1,371 Spanish sentences and 5,484 sentences total. Of the 1,371 Spanish sentences, we directly annotate 486, encompassing 9,540 words. There are five documents included in the Four Datasets dataset: Proxy reports from newswire data (Proxy), translated Xinhua newswire data (Xinhua), BOLT discussion forum source data (DFA), DARPA GALE weblog and Wall Street Journal data (Consensus), and BOLT discussion forum MT data (Bolt). For Consensus, Proxy, Bolt, and DFA, we annotate the first 100 sentences of the document. Xinhua is 86 sentences in total (averaging 22.37 words per sentence), so we annotate all 86 sentences. Consensus is originally 100 sentences (averaging 15.61 words per sentence), Proxy is originally 823 sentences (averaging 23.07 words per sentence), Bolt is 133 sentences (averaging 20.25 words per sentence), and DFA is 229 sentences long (averaging 17.83 words per sentence).

4.2 Annotator Training

Three undergraduate linguistics students, native English speakers with high levels of Spanish proficiency, were first trained in English AMR annotation. Annotators were then trained in our approach to Spanish AMR annotation, through discussions of our v1.0 Spanish AMR guidelines. *The Little Prince* corpus was used for practice annotation in both languages. Once trained, the annotators moved on to annotations of the Four Translations dataset. To verify annotator understanding, we completed adjudication on the test sets of English and Spanish annotations.

4.3 Collected Annotations

To validate our approach to annotation and the reliability of our annotations, we collect annotations from all three annotators for the first 50 sentences from the Proxy document. We are then able to perform inter-annotator agreement analysis on those overlapping annotations using Smatch, presented in §5. Other than those 50 Proxy annotations, all other annotations were distributed evenly between each of the three annotators. The three annotators produced 200, 190, and 196 annotations each. This results in a total of 586 annotations total, for 486 unique sentences, with Proxy 1–50 being annotated thrice (once by each annotator). After all annotations for the initial 50 sentences were produced, a final round of corrections were made for any errors in annotation (without changing any divergent judgment calls).

AMR annotation is expensive and time-consuming. Our 586 annotations took more than 200 hours to complete including some test annotations and correction of annotations. This is also a reflection of the sentences included in the AMR 2.0 - Four Translations dataset being especially difficult to annotate due to their complicated genre and length (approx. 20 words per sentence). To maximize the number of sentences with gold annotations, we refrained from double-annotating the remainder of the data beyond the aforementioned 50 sentences.

4.4 AnCora

We use the AnCora-Net Spanish lexicon of verbs (AnCoraVerb-ES) for verb sense annotation (Taulé et al., 2008). Similar to PropBank for English, the AnCora lexicon is comprised of predicates, accompanied by their argument structures and thematic roles. Each of the 2,647 predicate entries is also related to one or more semantic classes depending on its senses. AnCora also provides a lexicon of deverbal nominalizations, AnCoraNom-ES, which contains information regarding denotative type, WordNet Synset, argument structure, and the verb from which the noun is derived. As AnCoraNom-ES significantly overlaps with AnCoraVerb-ES, we choose not to use it in this work.

For all verbs or verb senses which did not appear in the AnCora corpus, we kept track of those instances in a table and supplemented the AnCora verb bank with 16 of our own. These added senses can be seen in table 1.

4.5 StreamSide Annotation Tool

Annotations were produced using the Streamside software (Choi and Williamson, 2021). The annotators annotate tokens in the sentence as concepts, and roles and arguments are then defined between these concepts as relations. While this software allows for annotation fitted to various languages, it is best accustomed to annotation using the English because the relevant PropBank roles (Kingsbury and Palmer, 2002; Palmer et al., 2005) are automatically populated. In our case, working on Spanish and using the AnCora rolesets (Taulé et al., 2008), the annotators needed to separately reference the arguments for each concept on the AnCora website.

4.6 Guidelines Development

We developed the guidelines by first outlining our approach to key Spanish linguistic phenomena, which we identified as potentially impacting Spanish AMR annotation. Our v1.0 guidelines discuss: (1) Use of English AMR Roles and Guidelines; (2) Pronoun Drop and NP Ellipsis; (3) Third Person Possessives; (4) *Se* Usage; (5) Gender; (6) Double Negation; (7) Diminutive and Augmentative Suffixes; (8) *Estar* (to be) as a Location.

These v1.0 guidelines were developed *before* performing any annotation. Since starting annotation, there have been 9 further iterations of the guidelines, which both expand on the items included in v1.0 and incorporate additional items. We discuss the most notable elements of the guidelines in §3. After developing the first iteration of the guidelines (v1.0), any further changes required to the guidelines, as identified during the annotation process, were incorporated into the next iteration. All existing annotations were then uniformly altered by their annotators to match the most updated guidelines.

5 Evaluation

5.1 Inter-Annotator Agreement

Table 2 shows the inter-annotator agreement (IAA) scores for each pair of annotators on the 50 triple-annotated Proxy sentences. The IAA scores were calculated by averaging the Smatch scores across the 50 sentence pairs for the annotators. The Smatch (Cai and Knight, 2013) algorithm calculates the amount of overlap between the AMR graphs to determine similarity. Smatch using a hill-climbing method to determine the optimal alignment between the variables in the AMR graphs and outputs an F-score from 0 to 1, where 1 indicates that the AMRs are isomorphic.

The average IAA scores ranged from 0.83–0.89, a very promising range for AMR annotation agreement. Comparable work achieved Smatch inter-annotator agreement scores of 0.79 (Choe et al., 2020), 0.72 (Sobrevilla Cabezudo and Pardo, 2019), and 0.83 (Li et al., 2016). Other work on cross-lingual AMR adaptations which only had one annotator did not report IAA/Smatch scores.

Ann. 1 & Ann. 2	0.89
Ann. 1 & Ann. 3	0.86
Ann. 2 & Ann. 3	0.83

Table 2: Average inter-annotator agreement scores (via Smatch) for each pair of our three annotators on the first 50 sentences of the Proxy document.

5.2 Disagreement Analysis

Disagreements, which we define as any discrepancy that neither violates AMR guidelines nor deviates from the sentence's meaning, were common among all three annotators. The majority of disagreements are caused by differences in interpretation.

Entity versus Event Annotation. AMR takes a predicate-centric approach to annotation. While verbs are typically annotated as events and nouns are annotated as entities (concepts without a number), when nouns or phrases have verbal counterparts, this can cause differences among annotators. For example, *propuesta* ("proposal") could be annotated either as a noun or as a verb (`proponer-01`, "to propose"). We instruct annotators to annotate derived nouns as verbs and annotate related roles as arguments for increased expressivity.

Verb Sense Labels. Verb senses account for nuance in meaning depending on context. Sometimes annotators chose different rolesets when the meaning difference between senses was subtle. One notable example is the verb *reconocer* ("to recognize / acknowledge"). `Reconocer-01` refers to recognizing something as official or true, as in *reconocer el estado* ("to recognize the state"). Alternatively, `reconocer-02` maintains that meaning, but often precedes a subordinate clause, as in *reconocen que gané* ("they acknowledge that I won").

Non-Core Role Overlap. Finally, annotators had difficulty consistently choosing the same non-core role (`:poss`, `:mod`, etc.) when the roles could overlap in meaning. For example, *la carta del hombre* ("the man's letter") could be annotated differently depending on the interpretation of the man's relationship to the letter. An emphasis on the man's ownership of the letter elicits the `:poss` role, whereas emphasizing the letter's creation by the man elicits the `:source` role.

t5wtense	0.7389
Fine-tuned t5wtense	0.8265
XLPT-AMR	0.8534

Table 3: BERTscore results for: the output of the t5wtense generation model without any fine-tuning, t5wtense after fine-tuning with our data, and the state-of-the-art XLPT-AMR cross-lingual AMR generation model (Xu et al., 2021) on our test split.

6 Fine-tuning a Spanish Generation Model

AMR generation produces text from an AMR. To evaluate the utility of our dataset in practical NLP tasks, we fine-tune the t5wtense generation model of the AMR library `amrlib` to produce Spanish sentences.[5] The t5wtense generation model uses the pretrained HuggingFace T5 transformer to convert AMR graphs to text. We split our 486 annotations into 110 sentences (test) and 376 (training).[6]

We compare the fine-tuned system output and the un-tuned system output to the corresponding Spanish reference sentences from AMR 2.0 - Four Translations (Damonte and Cohen, 2020). We use BERTScore, an automatic evaluation metric for text generation (Zhang et al., 2019), to perform this comparison, as previous work has demonstrated that it is the automatic metric most correlated with human judgments for (English) AMR-to-text generation systems (Manning et al., 2020).

For evaluating Spanish text, the default BERTscore model is bert-base-multilingual-cased, which is the model we use here. Table 3 shows AMR-to-Spanish BERTscore results.

After fine-tuning t5wtense, we see a marked improvement in performance, increasing in BERTscore by approximately 8.8% absolute (11.86% relative improvement). Current state-of-the-art cross-lingual generation (Xu et al., 2021) achieves a BERTscore of 0.8534 on the same test set,[7] which indicates that by fine-tuning on only 376 Spanish AMR annotations,

[5] `https://github.com/bjascob/amrlib`

[6] We split the data as follows: Training set: Bolt 1–100, Consensus 1–100, DFA 1–40, Proxy 51-100, Xinhua 1–86; Test set: DFA 41–100, Proxy 1–50.

[7] Xu et al. (2021) report SOTA scores using BLEU. We computed BERTscore on their system's output.

we are able to achieve results close to the current best performing model.[8] The marked improvement resulting from our fine-tuning demonstrates the utility of our corpus and suggests incorporating our data into more sophisticated generation or parsing models can lead to greater improvements.[9]

7 Conclusion

We have presented an updated approach to Spanish AMR annotation which considers a broader range of meaningful linguistic phenomena than previous work. Using updated guidelines, we constructed a corpus of 486 gold-annotated Spanish AMRs for the "AMR 2.0 - Four Translations" dataset, achieving high AMR inter-annotator agreement (0.83–0.89 IAA via Smatch). Gold Spanish AMRs will contribute to ongoing evaluation and training of cross-lingual AMR models; this is substantiated by our results in §6, which improved an off-the-shelf AMR-to-Spanish generation system by fine-tuning on our data. Little prior work on AMR has set out to develop large-scale gold corpora in languages other than English; our work suggests that this is a fruitful effort, both to foster a better understanding of the cross-lingual properties of AMR and to improve system performance on non-English NLP tasks.

Acknowledgements

We thank anonymous reviewers for helpful feedback. We also thank the coordinators of the Georgetown University RULE (Research-based Undergraduate Linguistics Experience) program. This work is supported by a Clare Boothe Luce Scholarship.

References

Rafael Anchiêta and Thiago Pardo. 2018. Towards AMR-BR: A SemBank for Brazilian Portuguese language. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

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, Sofia, Bulgaria. 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. 2019. Abstract Meaning Representation (AMR) 1.2.6 specification. `https://github.com/amrisi/amr-guidelines/blob/master/amr.md`.

Rexhina Blloshmi, Rocco Tripodi, and Roberto Navigli. 2020. XL-AMR: Enabling cross-lingual AMR parsing with transfer learning techniques. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2487–2500, Online. Association for Computational Linguistics.

Claire Bonial, Julia Bonn, Kathryn Conger, Jena D. Hwang, and Martha Palmer. 2014. PropBank: semantics of new predicate types. In *Proc. of LREC*, pages 3013–3019, Reykjavík, Iceland.

Deng Cai, Xin Li, Jackie Chun-Sing Ho, Lidong Bing, and Wai Lam. 2021. Multilingual AMR parsing with noisy knowledge distillation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2778–2789, Punta Cana, Dominican Republic. Association for Computational Linguistics.

[8] The Xu et al. (2021) system performance is hindered by the fact that two of the AMRs do not produce any output at all by this model. If we remove those two AMRs from consideration, the F1 score for the Xu et al. (2021) system is slightly higher, achieving a BERTScore F1 of 0.8695, while our fine-tuned results are 0.8266 on the same sentences.

[9] Xu et al. (2021) do not release their code, so the model cannot be fine-tuned.

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.

Hyonsu Choe, Jiyoon Han, Hyejin Park, Tae Hwan Oh, and Hansaem Kim. 2020. Building Korean Abstract Meaning Representation corpus. In *Proceedings of the Second International Workshop on Designing Meaning Representations*, pages 21–29, Barcelona Spain (online). Association for Computational Linguistics.

Jinho D. Choi and Gregor Williamson. 2021. Streamside: A fully-customizable open-source toolkit for efficient annotation of meaning representations.

Marco Damonte and Shay Cohen. 2020. Abstract Meaning Representation 2.0 - Four Translations. Technical Report LDC2020T07, Linguistic Data Consortium, Philadelphia, PA.

Lucia Elizabeth Donatelli. 2019. *The Morphosemantics of Spanish Gender: Evidence from Small Nominals*. Georgetown University.

RAE Real Academia Española. 2010. *Nueva gramática de la lengua española manual*. Espasa.

Jan Hajič, Ondřej Bojar, and Zdeňka Urešová. 2014. Comparing Czech and English AMRs. In *Proceedings of Workshop on Lexical and Grammatical Resources for Language Processing*, pages 55–64, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Johannes Heinecke and Anastasia Shimorina. 2022. Multilingual Abstract Meaning Representation for Celtic languages. In *Proceedings of the 4th Celtic Language Technology Workshop within LREC2022*, pages 1–6, Marseille, France. European Language Resources Association.

Marcio Lima Inácio, Marco Antonio Sobrevilla Cabezudo, Renata Ramisch, Ariani Di Felippo, and Thiago Alexandre Salgueiro Pardo. 2022. The AMR-PT corpus and the semantic annotation of challenging sentences from journalistic and opinion texts. *SciELO Preprints*.

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).

Kevin Knight, Bianca Badarau, Laura Banarescu, Claire Bonial, Madalina Bardocz, Kira Griffitt, Ulf Hermjakob, Daniel Marcu, Martha Palmer, Tim O'Gorman, and Nathan Schneider. 2017. Abstract Meaning Representation (AMR) Annotation Release 2.0. Technical Report LDC2017T10, Linguistic Data Consortium, Philadelphia, PA.

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.

Emma Manning, Shira Wein, and Nathan Schneider. 2020. A human evaluation of AMR-to-English generation systems. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4773–4786, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Noelia Migueles-Abraira. 2017. A study towards Spanish Abstract Meaning Representation. Master's thesis, University of the Basque Country.

Noelia Migueles-Abraira, Rodrigo Agerri, and Arantza Diaz de Ilarraza. 2018. Annotating Abstract Meaning Representations for Spanish. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Elif Oral, Ali Acar, and Gülşen Eryiğit. 2022. Abstract Meaning Representation of Turkish. *Natural Language Engineering*, page 1–30.

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

Luigi Procopio, Rocco Tripodi, and Roberto Navigli. 2021. SGL: Speaking the graph languages of semantic parsing via multilingual translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 325–337, Online. Association for Computational Linguistics.

Marco Antonio Sobrevilla Cabezudo and Thiago Pardo. 2019. Towards a general Abstract Meaning Representation corpus for Brazilian Portuguese. In *Proceedings of the 13th Linguistic Annotation Workshop*, pages 236–244, Florence, Italy. Association for Computational Linguistics.

Reza Takhshid, Razieh Shojaei, Zahra Azin, and Mohammad Bahrani. 2022. Persian Abstract Meaning Representation.

Mariona Taulé, M. Antònia Martí, and Marta Recasens. 2008. AnCora: Multilevel annotated corpora for Catalan and Spanish. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Shira Wein and Nathan Schneider. 2021. Classifying divergences in cross-lingual AMR pairs. In *Proceedings of The Joint 15th Linguistic Annotation Workshop (LAW) and 3rd Designing Meaning Representations (DMR) Workshop*, pages 56–65, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Dongqin Xu, Junhui Li, Muhua Zhu, Min Zhang, and Guodong Zhou. 2021. XLPT-AMR: Cross-lingual pre-training via multi-task learning for zero-shot AMR parsing and text generation. 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 896–907, Online. 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 Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1765–1772, Reykjavik, Iceland. European Language Resources Association (ELRA).

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. BERTScore: Evaluating text generation with BERT. *arXiv preprint arXiv:1904.09675*.

Part-of-Speech and Morphological Tagging of Algerian Judeo-Arabic

Ofra Tirosh-Becker[1*] Michal Kessler[2*] Oren M. Becker[3] Yonatan Belinkov[4†]

[1] The Hebrew University of Jerusalem, Israel. otirosh@mail.huji.ac.il
[2] The Hebrew University of Jerusalem, Israel. michalskessler@gmail.com
[3] Becker Consulting Ltd., Mevasseret Zion, Israel. becker.oren@gmail.com
[4] Technion – Israel Institute of Technology, Israel. belinkov@technion.ac.il

Abstract Most linguistic studies of Judeo-Arabic, the ensemble of dialects spoken and written by Jews in Arab lands, are qualitative in nature and rely on laborious manual annotation work, and are therefore limited in scale. In this work, we develop automatic methods for morpho-syntactic tagging of Algerian Judeo-Arabic texts published by Algerian Jews in the 19th–20th centuries, based on a linguistically tagged corpus. First, we describe our semi-automatic approach for preprocessing these texts. Then, we experiment with both an off-the-shelf morphological tagger, several specially designed neural network taggers, and a hybrid human-in-the-loop approach. Finally, we perform a real-world evaluation of new texts that were never tagged before in comparison with human expert annotators. Our experimental results demonstrate that these methods can dramatically speed up and improve the linguistic research pipeline, enabling linguists to study these dialects on a much greater scale.

1 Introduction

Application of Natural Language Processing (NLP) to real-world problems has been the field's goal from its early days. As algorithms advance, the contribution of NLP to real problems has become more evident and more substantial. The present study originates from a real-world challenge faced by linguists of Semitic languages, in this case researchers of the Judeo-Arabic dialects of Algeria (AJA). Their challenge, simply put, is how to scale up linguistic analyses of such dialects. Semitic languages in general, and Arabic in particular, are characterized by a very rich morphology that uses both templatic and concatenative morphemes, combined with the use of a vowelless script ("*abjad*"). This makes morphological analysis of Arabic very time-consuming even for expert linguists. Because speakers of the AJA dialects are becoming scarce, the attention of linguists in this field has shifted from fieldwork interviews with native speakers to library-based analysis of texts written in those dialects. Fortunately, vast collections of AJA texts were preserved in printed books, journals and handwritten manuscripts. Analyzing this linguistic treasure-trove, however, is proving to be challenging due to its size. The time-consuming manual annotation does not scale, and requires expertise that is hard to find.

We aim to scale up the linguistic analysis of this Arabic dialect using NLP tools. In particular, our goal is to develop an NLP tool that will assist AJA linguists in their *real-world task*, in a way that *they* will find it useful. Basing our work on the existing linguistically Tagged Algerian Judeo-Arabic (TAJA) corpus (Tirosh-Becker and Becker, 2022), we set out to develop automatic methods for morpho-syntactic tagging of such texts. Several specially designed neural network taggers and an off-the-shelf morphological tagger were experimented with, and assessed for their accuracy and likely usefulness. We also considered a hybrid human-in-the-loop approach. Finally, we carried out a *real-world evaluation* of our best performing part-of-speech (POS) taggers, applying them to untagged texts and assessing their quality via a user study with expert AJA linguists. Our experimental results demonstrate that these methods can dramatically speed up and improve the linguistic research pipeline, enabling linguists to study this language on a much greater scale.

[*] Equal contribution

[†] Supported by the Viterbi Fellowship in the Center for Computer Engineering at the Technion.

2 Linguistic Background

Judeo-Arabic (JA) lies in the intersection of Semitic languages and Jewish languages. As a Semitic language, and more specifically, an Arabic language variety, its words are generally composed of 3-letter roots, with added vowels and consonants according to pattern paradigms, as well as affixes and clitics (McCarthy, 1981). Arabic is the most widely spoken Semitic language, with 300 million native speakers (Owens, 2013). In fact, the term 'Arabic' refers both to Modern Standard Arabic (MSA) and to the Arabic dialects spoken throughout the Arab World. The two varieties of Arabic coexist in a state of diglossia (Ferguson, 1959) or continuuglossia (Hary, 2003), meaning the language varieties exist side by side, with writers or speakers shifting between varieties according to circumstance. MSA is written using the Arabic script, which is a right-to-left alphabet. Arabic dialects are usually written in Arabic script as well, but there is no standardized spelling for dialectal Arabic (Habash et al., 2012).

Arabic uses both templatic and concatenative morphemes. There are two types of templatic morphemes: roots and templates. Roots are usually three consonantal radicals that signify some abstract meaning. Roots are inserted into abstract patterns called templates.

There are two kinds of concatenative morphemes that attach to the templatic morphemes. Clitics are morphemes that have the syntactic characteristics of words, but are phonologically bound to another word (Zitouni, 2014), for example "*wa*",[1] meaning "and". Affixes are phonologically and syntactically part of the word, and often represent inflectional features, such as person, gender, number, and more.

Dialectal Arabic (DA) is a primarily spoken family of language varieties (and in modern days, widely used in written form on social media as well) that exist alongside the written MSA. DA diverges from MSA on several levels. There are differences in phonology, morphology, lexicon, and orthography (Habash et al., 2012). The regional dialects can be broken down into main groups, with one possible breakdown being Egyptian, Levantine, Gulf, Iraqi, and Maghrebi. Even within dialect groups there can be quite a lot of variance between dialects, although in many cases there is a certain level of intelligibility between speakers of different dialects, with more significant difficulty across dialect groups. Maghrebi dialects are influenced by the contact with French and Berber languages, and the Western-most varieties could be unintelligible by speakers from other regions in the Middle East, especially in spoken form (Zaidan and Callison-Burch, 2014).

[1] We use the Habash-Soudi-Buckwalter transliteration scheme (Habash et al., 2007) for Arabic text. For AJA texts, we use the common transliteration of JA; see Table 9 in the appendix.

While JA can be looked at as an ensemble of Arabic dialects, it is first and foremost a subgroup of Jewish languages. Jewish languages are a family of language varieties that developed in Jewish communities throughout the diaspora. The original language used by Jews in the Land of Israel was Hebrew, followed closely by Aramaic. As Jews spread across the world, they adopted local languages and developed distinctive varieties of these languages. Nonetheless Hebrew remained their liturgical language, even as it almost died out as a spoken language until its revival in the late 19th and early 20th centuries. Perhaps the most well-known of these Jewish languages is Yiddish, the Judeo-German language developed by Ashkenazi Jews living in Central and Eastern Europe before the Holocaust. Jewish languages vary in their distance and divergence from their non-Jewish sister languages, some being influenced by multiple languages due to language contact. Nonetheless, among the features that tie these languages together are the presence of Hebrew and Aramaic lexical components (Kahn and Rubin, 2017), the use of the Hebrew alphabet for writing, and more.

Algerian JA (AJA) is a member of the North African Judeo-Arabic dialect group, i.e., dialects spoken and written by Jews of the Maghreb. AJA is in contact with Moroccan and Tunisian Arabic dialects (both Jewish and Muslim), with French and to a lesser extent other trade languages such as Spanish and Italian, and with Hebrew and Aramaic, the historical Jewish cultural languages. In general AJA shares many characteristics with other Jewish languages, including the use of Hebrew script, presence of Hebrew and Aramaic components, and a mixture of conservative trends, vernacular features, and heterogeneous elements (Tirosh-Becker, 2012). To date, AJA has been sparsely studied by linguists. The AJA dialect of the city of Algiers was studied over a century ago by Cohen (1912), with most of the recent work on AJA published by Tirosh-Becker, focusing on Constantine, the third largest city in Algeria (Tirosh-Becker, 1988, 1989, 2011a,b, 2014). AJA research employs fieldwork interviews of informants and the study of selected written texts (e.g., Bar-Asher, 1992; Tedghi, 2012; Tirosh-Becker, 2011a,c, 2012). Regretfully, the number of AJA speakers has decreased following Algeria's independence (in 1962) and the subsequent dispersion of its Jewish communities, making fieldwork today almost impossible. Hence, this research is now shifting towards an analysis of the vast textual sources left by many of these Jewish communities, in both manuscript and print form. Most of the linguistic analyses done thus far on AJA texts have been based on single or few texts, as each study requires extended effort of poring over texts, dictionaries, and grammars. Given the size of these corpora, this is a perfect match for machine learning and NLP approaches.

3 Related Work

3.1 Arabic Corpora

Corpora for Arabic NLP are usually gathered with a specific language variety in mind, and optionally annotated with information for specific tasks. We briefly discuss here the most prominent and relevant Arabic corpora, and refer to Belinkov (2021) for a broader survey. Masader (Alyafeai et al., 2022; Altaher et al., 2022) is an online catalogue of Arabic NLP datasets.

The majority of annotated Arabic corpora are for MSA. The most prominent annotated MSA corpora are the Penn Arabic Treebank (PATB; Maamouri et al., 2004), and the Prague Arabic Dependency Treebank (PADT; Hajič et al., 2009), a dependency treebank for MSA. In addition, El-Haj and Koulali (2013) present KALIMAT, a multipurpose corpus for MSA, with over 20,000 articles and over 18 million words, annotated using existing state-of-the-art Arabic NLP tools for POS tags, morphological analyses, named entity recognition (NER), and auto-summarization.

There are also annotated corpora for DA. ATB-ARZ (Maamouri et al., 2014) is an Egyptian Arabic treebank, with 182,965 tokens after clitic splitting. This corpus is annotated for POS, morphology, gloss, and syntactic treebank, following the guidelines of the PATB. There are several corpora for dialect identification that include Algerian and other Maghrebi dialects, such as Habibi (El-Haj, 2020), a corpus of Arabic song lyrics, or QADI (Abdelali et al., 2021). Seddah et al. (2020) created the NArabizi corpus, North African Arabic written in Latin letters (commonly known as Arabizi), with 1500 sentences of annotated Algerian dialectal Arabic, with tokenization, morphological analysis, code-switching identification, syntactic annotations, and sentence-level translations in French. MADAR (Bouamor et al., 2018) has 12,000 sentences with parallel translations in mutliple dialects, including Algerian and other North African dialects, but without morphological annotations. In addition, Zribi et al. (2015) transcribed and annotated a spoken Tunisian Arabic corpus, a North African dialect that is close to Algerian Arabic. It is worth noting that many DA corpora are transcribed from audio sources and are not originally textual data.

As for Judeo-Arabic corpora, the only publicly available JA corpus to date is the Friedberg Judeo-Arabic Project,[2] with almost 4 million words from 110 pre-modern JA texts, including texts by Rav Saadia Gaon and Maimonides. The only annotation available for these words is language (Arabic, or Hebrew/Aramaic). Recently, Tirosh-Becker and Becker (2022) developed the TAJA (Tagged Algerian Judeo-Arabic) corpus, a linguistically annotated corpus of written Algerian Judeo-Arabic. This corpus is a collection of modern AJA texts published in Algeria in the late 19th and the first half of the 20th century. Section 4 provides a detailed description of the TAJA corpus, on which this paper is based.

[2] https://fjms.genizah.org/

3.2 Arabic POS Tagging and Morphological Analysis

Much of the work done on POS tagging in Arabic has used statistical methods. Diab (2009) uses an SVM classifier for choosing POS tags on MSA. MADAMIRA (Pasha et al., 2014), trained on the MSA PATB, is often used as a benchmark for Arabic POS tagging. It uses a morphological analysis component as part of the preprocessing stage, and then uses SVM and language models to predict POS tags, as well as tokenization, NER, and other tasks. Farasa is another Arabic NLP tool with support for POS tagging in MSA and DA, which is based on conditional random fields (Abdelali et al., 2016; Darwish et al., 2018). In recent work, deep neural networks have been used to train POS and morphological taggers. Plank et al. (2016) built POS taggers for 22 languages, including Arabic, using data from the Universal Dependencies project (Nivre et al., 2015). They experiment with using word embeddings, character embeddings, byte embeddings, and some combinations thereof. Their best performing model does especially well on Arabic, reaching up to 98.91% accuracy.

Works that cover DA often leverage tools developed on or for MSA. Duh and Kirchhoff (2005) propose a minimally supervised approach for POS tagging of DA that combines raw text data from several varieties of Arabic, and a morphological analyzer for MSA with no other dialect-specific tools. Habash et al. (2013) tweak the MSA morphological analyzer MADA (Roth et al., 2008) for analyzing Egyptian DA, rather than the original MSA. They achieve up to 84.5% accuracy on morphological tags and 90.1% on Penn POS tags.

Other studies that address both MSA and DA have used bi-LSTMs for morphological tagging, sometimes jointly with other tasks like diacritization (Zalmout and Habash, 2020, 2019). Very recently, Inoue et al. (2022) have shown benefits from using pre-trained Transformer language models, especially when transferring from high- to low-resource dialects or language varieties, outperforming previous approaches.

Darwish et al. (2020) introduce a robust multi-dialect POS tagging system trained on tweets from four different dialect groups. They implement two approaches: the first uses CRFs, and the second stacks layers of CNNs, recurrent neural networks (RNNs), and a CRF layer. Their dataset comprises hundreds of tweets in each dialect group, each manually segmented into tokens and clitics. They make use of stem templates and Brown clusters as features concatenated to the embeddings for classification, and achieve accuracy of up to 92.4% on the POS tagging of seen words and

source text reference	line number	context	word
Gn_avot_4:16	2	אדנייא האדי תשבאה לסקיפ׳א ʾdnyyʾ hʾdy tšbʾh lsqyfʿ	אדנייא ʾdnyyʾ
lemma/root	POS	morphological analysis 1	morphological analysis 2
דנייא dnyyʾ	noun	feminine	signular
additional tags	enclitic pronoun	comments	orthography and pronunciation
NA	NA		1. yy denotes consonantal ya' 2. Phonetic transcription of the definite article

Table 1: The general structure of a word-record in the TAJA corpus with a specific example. The words and context are stored in the word-record in their original Hebrew script; transliterations are added here for clarity.

82.9% on unseen words in Maghrebi dialects.

Given the orthographic, grammatical, and lexical differences between JA on the one side and MSA and other Arabic dialects on the other side, it is not straightforward to apply tools developed for MSA and Arabic-script DA to processing JA. Future work may investigate ways to transfer such tools or incorporate them with JA-dedicated tools. Efforts to transliterate JA texts to Arabic script (Terner et al., 2020) may assists in pursuing this direction.

3.3 Code-Switching

While there is no work known to us applying NLP to JA, there is work on code-switching, which is a significant characteristic of JA, as we noted in Section 2. Code-switching is when a speaker alternates between two or more languages or dialects in the context of a single conversation or situation. Ahmed (2018) annotates Hebrew elements in JA, capturing cases of code-switching, borrowing, and Hebrew quotations, and investigating sociolinguistic aspects in medieval JA texts. Wagner and Connolly (2018) perform a quantitative analysis of code-switching in JA texts from the Cairo Geniza.

As Çetinoğlu et al. (2016) point out, POS tagging of code-switched data is much harder than tagging monolingual texts, as models could reach 97% accuracy for the latter, but as low as 77% for the former. Attia et al. (2019) find that POS tags provide a strong signal for identifying code-switching. Just as code-switching is a major characteristic of AJA, it also characterizes other varieties of Algerian Arabic, and poses a challenge to Arabic NLP research (Riabi et al., 2021).

4 Data

This project has used the Tagged Algerian Judeo-Arabic (TAJA) corpus developed by Tirosh-Becker and Becker (2022).[3] This AJA corpus is a collection of modern AJA texts published in Algeria in the late 19th and the first half of the 20th century. The texts represent a variety of prose genres written by Algerian Jews, including:

- Bible translations, known as *šarḥ* (sg.) or *šurūḥ* (pl.).
- Translations of Hebrew post-biblical texts (such as the Mishnah, the Passover Haggadah, and liturgical poems).
- Translations of other Hebrew texts (such as Maimonides' *Mishne Torah*).
- Original writings composed in AJA, including commentaries and writings about Jewish law.
- Journalistic writings in AJA.

These texts were manually typed into computer-readable format and subsequently proofread, as Hebrew OCR (Optical Character Recognition) failed on these AJA texts. This was due not only to the less-than-favorable conditions under which the books had been stored, leaving the pages grayed and worn, but also because the fonts used in these books are not identical to standard Hebrew, as they have JA-specific adaptations, such as diacritics. Each text was manually tokenized and annotated by research assistants (RAs, usually MA or PhD candidates) in a spreadsheet, according to strict guidelines, and most were verified by a senior expert.

The digitization and annotation project spanned several years, with some dozen RAs contributing to the annotation efforts. Approximately 80% of the time spent on the creation of TAJA was dedicated to the annotation process, as the digitization is a more straightforward (though non-trivial) task.

4.1 Data Annotation

The TAJA corpus was created to be a *linguistically annotated* digital corpus of *genre-diverse written* texts (Tirosh-Becker and Becker, 2022). The basic elements in this digital corpus are the individual words. Generally speaking, the texts are split on white-spaces, though there are some multi-word expressions that are annotated as a single unit. Each word is stored in a sort of *word-record*, which places the word in its sentence-level context (as well as a reference to the full text), and provides linguistic information about its grammatical components and more (Table 1).

[3]The corpus is available through the authors.

4.1.1 Parts of Speech (POS)

Each word is tagged with a unique POS tag. The tags are drawn from a closed list of the following POS: noun, verb, particle, proper noun, relative pronoun, adjective, number, personal pronoun, demonstrative, adverb, presentative, quantifier, and acronym. POS tagging was also applied to the embedded Hebrew, Aramaic, and French words, which are identified in the TAJA corpus by code-switching tags, as these embedded words are interwoven into the syntactic fabric of JA. In almost all cases these code-switching words were nouns. See Table 10 (Appendix) for a list of valid POS tags.

4.1.2 Morphological Tags

The morphology of each word was fully analyzed by expert JA linguists. Each POS tag calls for its own set of morphological features. Given a noun, for example, we expect information about gender, number, and code-switching. The fields in our dataset in which we find this morphological information are *analysis1*, *analysis2*, *additional tags* and *enclitic pronouns*. Note that there is a clear ranking between these fields. Most of the morphological information is captured by the first two fields, *analysis1* and *analysis2*, reflecting the rich morphology of AJA, while the *additional tags* field refers to a small subset of morphological attributes that apply only to a limited number of POS tags, i.e., to verbs (combinations of person, gender, and number) or to demonstratives (proximal vs. distal). The information provided by the *enclitic pronouns* field is morphologically more restricted. Each POS tag generally has its own set of legal values for these analyses, and they do not often overlap with the legal morphological annotations of other POS tags. In fact, at times, the same linguistic information may appear in different annotation fields for different POS tags. For example, code-switching information for nouns appears in the *analysis1* field, but the same information for proper nouns appears in *analysis2*. See Tables 11–14 (Appendix) for lists of valid morphological tags for the prominent POS tags.

4.2 Corpus Statistics

TAJA is comprised of 69 spreadsheet files, which cover 16 printed texts. These include 9904 AJA sentences, with a total of 61,481 tokens. There are 17,876 different word types in the corpus, for a type–token ratio (TTR) of 0.2907. It is important to recall that AJA is a highly morphological language, with extensive use of affixes and clitics. For example, a word is marked as definite using the prefix *ʾl-*. The same is true for several prepositions, such as *b-* ("in" or "at") or *l-* ("to" or "for"). Thus, a single lemma with two different prefixes will be counted as two distinct word types, so the reported number of word types in fact represents fewer lemmas.

	Tokens	Types
Surface	20.89%	37.37%
Lemmas	5.34%	16.88%

Table 2: Out-of-vocabulary percentages for tokens and types, by surface level words and for lemmas.

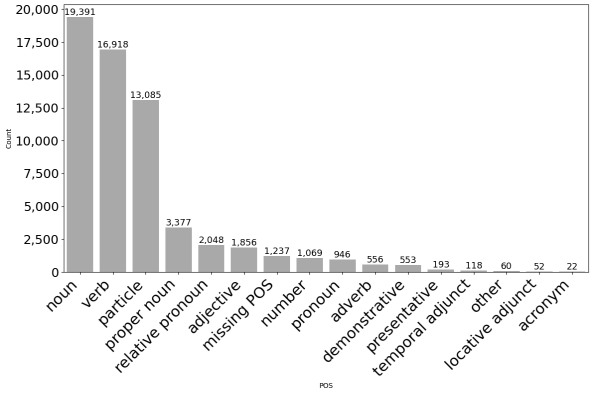


Figure 1: Part-of-speech tag distribution for the TAJA corpus.

As for the use of the term "lemma", the verbs in TAJA are tagged for the root rather than their lemma. In addition, approximately 2.1% of words in TAJA are missing the annotation in the lemma field, and are therefore left out of statistical calculations we report below at the level of lemmas. These issues limit our ability to provide accurate statistics on a lemma level.

For the 90/10 training/test split of TAJA with which we work in our experiments, we see high out of vocabulary (OOV) percentages for surface-level words (Table 2). When looking at word types, we see that more than a third of surface-level word types in the test set did not appear in the training set. Recall that this includes words that appear in the training set with an affix (such as a determiner, for example), and appear in the test set without said affix (or vice versa). We also look at the lemma OOV percentage, despite what we explained above about verbs being annotated for root instead of lemma. There is a large portion of OOV lemma types in the test set. These characteristics illustrate the diversity of the data in both lexical and surface form levels.

Finally, the data suffer from a long-tailed distribution of the annotations, a common problem in NLP. When examining the distribution of POS tags, for example, the three most common POS tags (noun, verb, and particle) account for approximately 80% of the annotations (see Figure 1), while the other 11 valid POS tags comprise only a fifth of the annotations.

4.3 Corpus Ambiguity

Before we discuss the ambiguity statistics of the TAJA corpus, we must address the noisiness of the data. Despite the laborious annotation effort (Tirosh-Becker and Becker, 2022), the data still contain problematic annotations. We discuss our attempts to clean the data below, but at this point, it is enough to know that some annotations appear with typos, or with additions that should not be there, such as question marks.

Corpus ambiguity is defined in Dermatas and Kokkinakis (1995) as the mean number of possible tags for each word of the corpus. This number can provide a signal for the difficulty of the tagging task. The corpus ambiguity of TAJA, calculated on the surface-level tokens, is 1.7497. This is high relative to the corpus ambiguity of other language corpora, as reported in Dermatas and Kokkinakis (1995), which range from 1.11 (for Dutch) to 1.69 (for French). However, as we mentioned, the noise in the annotations makes this number unreliable. For example, if a word that appears many times in the corpus appears one time with a typo in the annotation, this will raise the corpus ambiguity unjustifiably.

4.4 End-Goal: The Unannotated NAJA Corpus

In addition to TAJA, there exists a larger unannotated corpus of digitized AJA text. The New Algerian Judeo-Arabic (NAJA) corpus includes the same genres as TAJA, though differently distributed. The estimated size of NAJA is between 170k-186k tokens, almost three times as many as TAJA. It is the laborious task of manually annotating this corpus that we wish to automate away, using taggers trained on TAJA.

5 Preprocessing

In this section, we describe several challenges we faced in the preprocessing stage and the steps we took to address them.

5.1 Invalid annotations

Although the annotators were provided with the list of legal tags and legal morphological annotations, the data are rife with 'illegal' values, including mistyped tags (e.g., צל instead of מל),[4] two tags combined into one (מל+שע rather than שע), annotations with question marks and slashes (indicating that they are not confident about the tag they chose) and most often, words that are simply missing a tag.

[4]This mistyping is caused by the letter צ (ṣ) being adjacent to the letter מ (m) on the Hebrew keyboard. Table 10 (Appendix) provides the list of POS codes and their meanings.

We took a semi-automated approach for correcting as many annotations as possible. We created a mapping from misspelled or mistyped tags to the correct spelling. This was an iterative process, as at each iteration new categories of errors emerged, requiring additional consultation with JA language experts. For example, when resolving combined tags (as we described above), it is not obvious that it is desirable to drop the information represented by either of the tags. Being able to automate away the correct or obvious cases, enabled us to narrow down the number of questions we needed to bring to the experts, and conversely, having a language expert to whom we could bring the difficult questions, allowed us to ensure the annotations are as accurate as possible. Upon loading the spreadsheets and ingesting the data, we automatically convert any incorrect tags that appear in our mappings to the correct tags. These mappings catch 662 errors that are automatically corrected as part of the preprocessing stage. Using regular expressions we collected the cases of low confidence annotations (indicated by question marks or slashes in the original spreadsheets), and sent them for review by the language experts. Most of these were corrected manually in the original spreadsheets, in addition to some errors found in the enclitic pronouns, for a total of 64 manual corrections. Finally, missing annotations are represented with an underscore.

5.2 Column offsets

Another kind of noise we encountered in the annotated data are column offsets (e.g., the POS tag appears in the *analysis1* column, and so on). During preprocessing, we check automatically for such offsets in the columns, and automatically realign the annotations to their correct fields while parsing. We found 64 such cases, and fixed them automatically.

5.3 Multi-word expressions

The spreadsheet input includes the tokenization of each sentence, listing each token on a separate line, where each sentence is separated from the next by an empty line. In most cases, the tokenization is done on whitespaces. However, on various occasions, a multi-word expression appears on a single line and is annotated as a single unit. This happens most commonly with proper nouns, such as ראש השנה (*rʾš hšnh*, 'the New Year'; this is also a Hebrew construct phrase) or יהושע ולד נון (*yhwšʿ wld nwn*, 'Joshua son of Nun'), or Hebrew phrases such as the phrase בעולם הזה (*bʿwlm hzh*, 'in this world'; includes a noun) or ועשה טוב (*wʿsh ṭwv*, 'and do good'; includes a verb). These multi-word expressions are most often Hebrew phrases or terms that are embedded in the AJA text. They are treated as a single word in TAJA, because they represent a single concept or entity. How-

ever, this potentially poses a problem, as the tokenization we perform on new texts is based on white-spaces and punctuation, and therefore when coming to annotate previously unseen texts with multi-word expressions, the tagger will address each component of the phrase as its own word, and it might be considered 'out of vocabulary' as far as our tagger is concerned. However, this is a very rare phenomenon, with fewer than 100 appearances in the entire corpus, and therefore we did not split multi-word phrases in our experiments.

6 Methods

6.1 The Tasks

We formulate both part-of-speech (POS) and morphological tagging as sequence labeling tasks. In the POS tagging task, we are given an input sentence of n words, denoted by $\mathrm{x} = w_1, \ldots, w_n$, and need to find the correct sequence of tags $\mathrm{t} = t_1, \ldots, t_n$, where t_i is taken from the set of POS tags T (Table 10, Appendix). Morphological tagging is performed on the same input as the POS tagging task. In this task, there are four morphological fields (*analysis1*, *analysis2*, *additional tags*, *enclitic pronoun*) to be tagged in addition to the POS tag field: $\mathrm{t}^1 = t_1^1, \ldots, t_n^1$, $\mathrm{t}^2 = t_1^2, \ldots, t_n^2$, $\mathrm{t}^3 = t_1^3, \ldots, t_n^3$, $\mathrm{t}^4 = t_1^4, \ldots, t_n^4$. Tables 11–14 (Appendix) contain lists of valid morphological tags for the prominent POS tags.

6.2 Models

We experiment with two types of models for the sequence labeling tasks: CRFs and RNNs. CRFs (Lafferty et al., 2001) are a framework for building probabilistic models for segmenting and labeling sequence data, while relaxing strong independence assumptions made by hidden Markov models (HMMs), and avoiding certain biases that maximum entropy Markov models (MEMMs) are prone to have. Parameter estimation is done by maximum likelihood estimation and the Viterbi algorithm is used for inference. CRFs use handcrafted features, such as the preceding and succeeding words, prefixes and suffixes, and more. In this study we experiment with MarMoT, an off-the-shelf tool that implements a pruned CRF model which has performed well on Modern Standard Arabic (Müller et al., 2013).

In addition to the standard MarMoT tool, we implement our own tagging model based on long short-term memory networks (LSTM; Hochreiter and Schmidhuber, 1997), a type of RNN that is more robust to the vanishing gradients problem and performs well on sequence-level tasks. Our backbone is a bi-directional LSTM model based on the PyTorch implementation (Paszke et al., 2019). On top of that, we add a linear layer that maps the hidden representations to the output space: either the space of all POS tags or the space of each of the morphological tag classes. Below we describe several improvements to this basic architecture.

6.3 Word-based vs. Character-based

Our basic LSTM architecture receives a sentence as input, and, using an embedding matrix for the words, passes the word embedding vectors $x_1 \ldots x_n$ through the LSTM one after another. However, this method has no way to deal with out-of-vocabulary (OOV) words, which are all mapped to a single 'UNKNOWN' token, and therefore to the same word embedding. It must use contextual information alone from neighboring words. OOV words are especially common in morphologically rich languages like AJA, as is evident from the corpus statistics (Section 4.2). To account for the highly morphological nature of AJA, it is important to address the characters on an individual level, as has been shown for other languages (Dos Santos and Zadrozny, 2014; Ling et al., 2015; Ballesteros et al., 2015). Looking at characters separately from words helps tag OOV words, because we can identify certain affixes that provide a strong signal about one of the annotations. For example, words starting with אל (ʾl), א (ʾ), or ל (l)[5] are more likely to be nouns.

For this purpose, we created two character-aware models. Both models train embeddings for the characters, but use different methods to create a word representation given the character embeddings.[6] Let the k^{th} word of sentence x be $w_k = c_{k,1}, c_{k,2}, \ldots, c_{k,m}$ (for ease of notation, $c_{k,i}$ represents both characters and character embeddings). The first method builds on the idea proposed by (Luong and Manning, 2016), and passes each word w_k's characters through an inner character-LSTM. The final hidden state $h_{k,m}$ of the character-LSTM is a character-aware word representation, which is concatenated to that word's embedding x_k. The combined representation $\tilde{x}_k = (x_k, h_{k,m})$ is fed to the word-level LSTM. We call this model CHAR-LSTM.

The second method follows (Kim et al., 2016), and uses a one-dimensional convolutional neural network (CNN), with a hyperparametric number of kernels K that convolve with the matrix of each word w_k's character embeddings. We apply a $\tanh$ non-linearity to the convolution outputs, and then pool the maximal values of each output to create a single character-based representation for each word, h_k. This representation is concatenated to the word's embedding. The combined representation $\tilde{x}_k = (x_k, h_k)$ is fed to the word-level LSTM. We call this model CNN.

[5] All these forms are related to the determiner אל (ʾl).

[6] One could use pre-trained word or character embeddings, but given the relatively small size of our corpus, we do not expect this to yield substantial improvements.

6.4 Flat vs. Hierarchical vs. Multitask Learning

Our basic experimental setup is to train tagging models for each field alone, resulting in five separate models (one for POS tagging and four for the morphological fields). We consider this 'flat' tagging a sort of baseline, as we hypothesize that including information from one field can improve results when predicting another.

The next setting we explore is a hierarchical model, utilizing a simple two-tier hierarchy with POS tags at the base and morphological tags building on that. This is anchored in the tag distribution. As mentioned in Section 4.1.2, most POS tags have their own set of legal morphological analyses in each field that are not shared with other POS tags. Thus, given the POS tag for a given word, the size of the possible pool of tags in each morphological field significantly decreases. Let $\tilde{x}_k$ be the word representation of word w_k including character information, as discussed above. In this setup, we also train five separate models, but while the POS model is identical to the base model architecture, the four morphological models concatenate POS tag information to the word representations, in the form of a one-hot vector $e_{t_k} \in \{0,1\}^d$ (where t_k is the index of the POS of w_k, for some ordering of all the POS tags, and d is the size of the POS tag set). The concatenated vector $(\tilde{x}_k, e_{t_k})$ is then fed to the word-level LSTM. During training, we provide the ground truth POS tag. At inference, we use POS tags predicted by the POS tagging model.[7]

Finally, a natural approach to take when tackling several tasks that are related to one another is multitask learning (MTL; Caruana, 1997), which has previous been considered for MSA morphological tagging (Inoue et al., 2017). In this setup, we share all parameters (word and character embeddings, and hidden states) between the different tasks, except for the final linear layer that receives the hidden states as input, and returns the scores for the relevant tag space. We have one layer of this kind for each task, each with its own parameters. We average the losses of each task, and backpropogate based on the averaged loss.

7 Experiments

7.1 POS Experiments

We begin our experimentation with addressing the POS tagging task alone, in order to determine the best architecture for our base model on a simpler task before diving into the more complicated morphology task. Our initial experiments are run with a base configuration of hyperparameters loosely based on prior work (Kim et al., 2016) and general intuition. Then we conduct a hyperparameter search for the best configuration. The exact settings are provided in Appendix B.

We run all our experiments by training on 90% of the tagged data, of which we hold out 10% for early stopping of the NN model training, and testing on the remaining 10%. All results of the neural-network-based models are averaged over five runs using five different seeds, unless noted otherwise. We compare the various model results to a 'most-frequent baseline' assignment, in which we assign a word the POS with which it appears most often in the training data, and assign all OOV words the most common POS tag (noun).

Table 3 summarizes the results of the various POS tagging models. The most frequent tag baseline is quite strong, as common in POS tagging tasks. In fact, it outperforms the WORD-LSTM model. Using character information is beneficial, and the CHAR-CNN model is better able to do so than the CHAR-LSTM model. Among the neural network models, it performs best. The best performing tagger overall is the CRF-based MarMoT tool.

Model	Accuracy [%]
most frequent baseline	82.01
WORD-LSTM	78.08±1.10
CHAR-LSTM	84.42±0.80
CHAR-CNN	87.45±0.58
MarMoT	89.17

Table 3: Accuracy of the POS tagging models. Best scoring model appears in bold.

7.2 Interim Summary

We saw in our experiments above that, among our neural-network (NN) approaches, representing a word by a CNN on its characters performs better than an LSTM, or ignoring the characters altogether. We use this CHAR-CNN model for hyperparameter tuning (see Appendix B). However, we also saw that MarMoT is indeed a very strong tool, and outperforms the CHAR-CNN in this task. Therefore, we move forward to the morphological tagging using both models, the CHAR-CNN representing the NN family, and MarMoT as a strong off-the-shelf tool.

7.3 Morphology Experiments

As we just discussed, of the three neural network architectures, the CHAR-CNN model performs best, and therefore we choose this architecture as our base model as we move forward with the morphology experiments,

[7] We use this setup for simplicity and do not consider curriculum learning strategies that sample targets both from the ground truth and from the model's predictions (Zhang et al., 2019).

Model	morphology			
	analysis1	analysis2	additional tags	enclitic
most frequent baseline	72.89	76.71	87.16	94.47
CHAR-CNN				
flat	80.18±0.47	84.02±0.53	90.59±0.08	95.72±0.10
hierarchical (pred POS)	79.56±0.32	83.69±0.76	90.05±0.53	95.87±0.14
hierarchical (true POS)	88.35±0.39	92.34±0.19	94.81±0.11	96.30±0.21
MTL	78.15±0.78	83.57±0.52	89.75±0.26	94.96±0.28
MarMoT	82.32	85.55	91.69	96.38

Table 4: Morphological models results by field. Best scoring results are in bold.

using the same base configuration of hyperparameters that we used in the POS experiments. We experiment with three approaches for predicting morphological tags (Section 6.4): the flat approach trains one model per each morphological attribute, the hierarchical approach uses POS information when predicting morphology, and the multitask approach predicts all morphological attributes jointly in a multitask manner. The hierarchical model was tested in two different setups, using either the predicted POS tag or the true POS tag in order to predict the morphological tags.

Providing true POS tags is a realistic choice for a linguistic annotation pipeline, since POS annotation is much simpler than morphological annotation. One may envision a human-in-the-loop process, where humans correct initial automatically assigned POS tags, and then a morphological tagger relies on the human-corrected tags. We return to this point in the discussion (Section 9).

7.3.1 Field by Field Accuracy

Table 4 shows the morphological tagging results broken down by field. The comparison highlights that our 'hierarchical CHAR-CNN model', when based on true POS, outperformed MarMoT in the first three morphology analysis fields. The model's success ranged from almost 89% for the *analysis1* field, almost 93% for *analysis2* and almost 95% for *additional tags*. This was judged as very significant by our JA experts. Due to its morphological complexity, manually tagging these morphology fields is highly time consuming even for experienced linguists. *Enclitic pronouns*, which are morphologically more restricted, are successfully predicted by most models with an accuracy greater than 95%.

7.3.2 Overall Accuracy

We also present several alternative overall scores for each of the taggers (Table 5). The 'strict' score considers a word to be correctly tagged only if all five fields are correctly tagged. This score was judged by the JA linguists as too severe, as they see real-world usefulness even if not all of the analysis fields were correctly tagged. The 'flexible' score counts each correct tag separately and gives equal weight to each field. Finally, reflecting the importance that our JA experts assigned to each field, a 'weighted' score was calculated as well, where the vector (3, 2, 2, 1, 1), for example, emphasizes POS over *analysis1* and *analysis2*, and gives the lowest weight to the *additional tags* and the enclitic pronouns. The comparison shows that our hierarchical CHAR-CNN (true POS) model performs better than MarMoT by 2.2% and 2.8% when calculating the 'flexible' score and the 'weighted' score, respectively, while MarMoT excels by the 'strict' metric.

Model	strict	flexible	weighted
most freq	66.94	82.64	80.76
CHAR-CNN			
flat	66.71±0.34	87.58±0.07	86.32±0.08
hierarchical (pred POS)	70.91±0.67	87.35±0.35	86.12±0.40
hierarchical (true POS)	71.18±0.52	91.95±0.13	90.71±0.15
MTL	66.24±0.97	86.84±0.30	85.72±0.30
MarMoT	75.84	89.02	87.92

Table 5: Overall accuracy scores for the morphological models. The strict, flexible, and weighted (3,2,2,1,1) scores are defined in the text.

7.3.3 Accuracy for words with legal tag combinations

Another way to evaluate our results is to look for all the words for which we know the tagger went wrong somehow. Recall that each POS has a certain set of legal values in each morphological analysis field, which differs from POS to POS (some of which can be seen in Tables 11–14 (Appendix). As our taggers are given the entire tagset, regardless of each specific word's POS, they may

Model	legal tag combo accuracy [%]	illegal tag combo average no. words	illegal tag combo percent [%]
CHAR-CNN			
flat	92.40±0.15	1652.0±44.9	26.95±0.73
hierarchical (pred POS)	88.56±0.20	1082.2±81.5	17.65±1.32
hierarchical (true POS)	95.18±0.25	1070.4±57.6	17.46±0.94
MTL	89.92±0.27	1379.0±44.6	22.49±0.73
MarMoT	89.49	1003.0	16.36

Table 6: 'Flexible' model accuracy for words with legal tag combinations, after removing words flagged as illegal combinations of POS tag and morphological analyses, and the average number of illegally tagged words and their percentage of the test set.

		morphology			
Model	POS	analysis1	analysis2	additional tags	enclitic
CHAR-CNN					
flat	72.63±0.52	59.64±0.99	61.06±1.28	73.80±0.61	88.40±0.43
hier. (pred)	72.58±0.46	57.72±0.81	59.64±1.39	72.26±1.85	89.01±0.68
hier. (true)	73.96±0.61	74.91±0.54	78.42±1.09	85.39±0.33	90.87±0.49
MTL	72.86±1.03	55.33±1.60	58.59±1.53	70.98±1.25	87.20±0.70
MarMoT	71.35	55.82	59.95	75.02	89.23

Table 7: Accuracy of morphology tagging for Out of Vocabulary (OOV) words.

produce illegal tag combinations if one of the predicted morphological tags does not appear in the legal values of the word's predicted POS tag (or, in the case of the true-POS-based hierarchical model, the true POS). In Table 6, we show the 'flexible' accuracy for each model on all the words that have legal tag combinations. Note that the accuracy of the true-POS hierarchical model for such words is almost 4% higher than its performance on the entire test set.

The table also shows the number of words that were tagged with illegal tag combinations and their percentage in the test set. Several observations can be made on the basis of this analysis. First, the models with the highest percentage of illegally tagged words are the flat CHAR-CNN and the multitask model. While the reported percentage of illegally tagged words for the true-POS-based hierarchical model (17%) is slightly higher than that of MarMoT, it is within a standard deviation of the percentage of words flagged in the MarMoT run. Coupled with the significant improvement in the 'flexible' score over MarMoT, which hardly improves over its general accuracy, this is a strong indication of the benefits of the true-POS-based hierarchical model.

We concede that 17% of all words is too many to expect a JA expert to address when using an automatic system for tagging new and unannotated data; however, these findings could potentially be used in other ways as well, such as adding a step in the automatic tagging process that forces the tagger to select a legal combination of POS tag and morphological analyses, using some heuristic to determine which of the predicted annotations to follow. This being said, as we mentioned in Section 4.3, the annotations in TAJA are noisy, and as such, 12% of the words in the annotated corpus appear with invalid analyses (mostly missing analyses, some illegal combinations) to begin with.

7.3.4 Out of Vocabulary Accuracy

Another way to evaluate how useful each model is in a real-world setting is through the accuracy of morphology tagging of Out of Vocabulary (OOV) words (Table 7) – words in the test set that did not occur in the training set. OOV words accounted for 21% of the TAJA test set (1281 of 6131 words). This high percentage of OOV words is reflective of the corpus' characteristics as discussed in Section 4.2. The results of this analysis are remarkable, with the hierarchical CHAR-CNN (true POS) significantly outperforming all other models across the different morphological analysis fields by 19% for *analysis1* and *analysis2* and by 10% for *additional tags*. This is a significant and encouraging finding, because it is very likely that the percentage of OOV words will increase in the future when we apply these tools to new texts beyond the TAJA corpus, especially if these texts are of different literary genres. Despite performing well in some of the previous evaluations, MarMoT failed on the morphology analysis of OOV words. A related ob-

servation is that even the hierarchical CHAR-CNN (predicted POS) model was able to assign POS tags to OOV words slightly better than MarMoT, achieving an accuracy of 72.58% vs. MarMoT's 71.35%.

8 Real-World Evaluation

The end goal of this project is to provide AJA language experts with an automatic tagger to help them annotate large volumes of text, a task which is otherwise laborious and time-consuming when tackled manually. To evaluate such real-world usefulness of the taggers we set out to compare the performance of our two best POS models (the hierarchical CHAR-CNN based model and MarMoT) with that of manual annotation by two expert AJA linguists.

8.1 The Task

For this evaluation task we selected a subset of the above-mentioned NAJA corpus, encompassing 30 chapters from the AJA translation of Psalms that were never annotated (a total of 3817 words). The two models were first trained on the entire TAJA corpus, and then we used the models to tag these selected unannotated texts. The resulting POS predictions were then given to two AJA experts of different calibers (see below), to evaluate and score. The two experts were instructed to write corrections only if one or both of the models were wrong, and to leave the annotation blank if both were correct. This enabled us not only to evaluate the performance of our competing models, but also assess inter-annotator agreement (IAA).

8.2 Inter-Annotator Agreement

It should be noted that the two human annotators that performed this task were of different expertise levels. One annotator is a senior professor of Judeo-Arabic, with decades of experience annotating and analyzing Judeo-Arabic texts. The second annotator is a doctoral student, a research assistant who has worked for several years under that professor's tutelage. Therefore, we consider the annotations of the senior expert to be the gold-standard, whereas the annotation of the junior expert is considered to be a silver-standard.

We calculate Cohen's Kappa between the annotations of the senior expert and those provided by the junior expert, excluding all the words on which the models disagreed but one of the human annotators did not identify the correct tag. We are left with 3685 words, for which $\kappa = 0.875$.

Note, however, that while Cohen's Kappa is a symmetric score, our two human annotators are of different calibers. Hence we take the senior expert's annotation as the correct result (gold-standard), and measure the accuracy of the junior expert's annotation relative to that of the senior expert. This will later be compared to the accuracy of the automatic taggers. Calculated on the same 3685 words stated above, the junior expert's accuracy in the wild was 0.908.

8.3 POS Tagger Evaluation

The accuracy statistics for the two POS taggers was evaluated relative to the corrections of the senior expert, whose annotations are considered to be the correct ones. Despite the instruction to correct all cases where at least one of the taggers was mistaken, there were 32 cases (0.8%) where the two models disagreed, but no correction was provided. On the remaining 3785 words, the accuracy of the two models was almost the same and only slightly lower than the accuracy of the human junior expert (Table 8). The real-world usefulness of the automatic taggers is highlighted when taking into account that it took the junior expert approximately 5.5 hours to complete this relatively limited task.

MarMoT	CHAR-CNN	junior expert
88.85	88.92	90.80

Table 8: POS tagging accuracy, on Psalms 1-30, relative to the correct tagging by the senior expert.

These results can be interpreted in several ways. A favorable way to look at this is that the automatic models are almost as good as a medium-level human annotator, and are therefore invaluable to the effort of annotating large amounts of text. A less favorable view is that a less experienced human annotator is more susceptible to agree with subtle mistakes made by an automatic tagger, though they might provide the correct annotation when facing a blank page. The easiest way to confirm or reject the hypothesis that the RA is more susceptible to being led astray by the automatic annotations is to compare his accuracy on this Psalms file to a similar number of annotations he made on a completely unannotated file. Unfortunately, that breakdown is not available. However, in support of this hypothesis, we break down the mistakes made by the junior expert by whether or not the models agreed on the annotation. We see that over 75% of the junior expert's mistakes were in cases where the models agreed, and of those cases, over 70% are words where the junior expert agreed with the automatic taggers, whereas the senior expert chose a different tag. In light of these numbers, it is important to emphasize to human annotators who use the automatically generated tags that they must look at the tags with a critical eye, and not assume that the taggers "know" the truth.

As is apparent from the results, there is almost no difference in accuracy between the two models, despite the fact that the models disagree on 11.2% of annotations. The number of mistakes made by each of the models is almost equal, with MarMoT being correct on 179 words and CHAR-CNN on 182 words out of 429 words on which the models disagree, and an additional 35 words on which both models were wrong. An interesting direction for future research is to characterize the kinds of mistakes each model tends to make, and explore ways to combine their strengths. Furthermore, we note that in this real-world application to NAJA (i.e., texts that are not part of TAJA) the CHAR-CNN model performed a little better than its initial TAJA-based evaluation (88.92% vs. 87.45%, see Table 3) while the MarMoT model performed a little worse (88.85% vs. 89.17%).

9 Discussion and Conclusion

The pressing *real-world* challenge facing researchers of Algerian Judeo Arabic (AJA) dialects is how to scale up their linguistic analyses from individual texts to large textual collections. The rich morphology of Arabic (as of other Semitic languages) and scarcity of expert linguists makes this complex and time-consuming task impractical unless aided by automation. Hence, developing automatic taggers that would support *real-world* linguistic analysis at scale and prove *useful* for AJA linguists is the challenge we aim to tackle. Reflecting the linguists' challenges, we focus on the performance of the morphological tagger in tests that are predictive of the real-world setting. For this reason, we did not limit ourselves to purely automated approaches, but also explored a hybrid human–machine approach, wherein the human expert contributes to the automatic approach.

The rich morphology of Arabic and its use of morpho-syntactic affixes led us to focus on character-based models (rather than word-based models), as these can identify key morphemes that are essential for annotating OOV words. Starting from a word-based LSTM neural network architecture, we integrated character-level information via either an LSTM or a CNN. Subsequently we explored a two-tier hierarchical approach to morphological tagging with POS tags at its base and the morphology tags building on that. This hierarchy mirrors the underlying character of Arabic annotation, where each POS tag has a set of legal morphological tags. The two-tier approach also enables exploring a human-in-the-loop step in between the two tiers. Our best performing strategy, denoted AJATAG for simplicity, is now available for use by AJA linguists.[8] To evaluate the *usefulness* of the AJATAG strategy we compared it to the *off-the-shelf* POS and morphological tagger, MarMoT, which is based on CRF. All models were trained on the annotated TAJA corpus.

For the base task of POS tagging, we found that among the evaluated neural network architectures, representing a word using a CNN run on its characters performed better than an LSTM or ignoring the characters altogether. Training on the TAJA corpus, the POS accuracy of the CHAR-CNN model was $87.4 \pm 0.58\%$. This accuracy is only slightly lower than the 89.17% accuracy obtained by MarMoT for this task. The 1.5% difference suggest essentially similar performance for the two models in a real-world setting. Morphology tagging, as indicated above, is the most challenging and time-consuming task that takes up 80% of the expert linguist annotation time. Here, too, CHAR-CNN performed better than the other neural network models we explored, especially in a two-tier hierarchical approach. The accuracy of this model, denoted herein as 'hierarchical CHAR-CNN (predicted POS)', ranges from 81% to 91% for the different morphology analysis fields (*analysis1*, *analysis2*, *additional tags*). To further improve the performance, we allowed for human input between the two tiers in the form of manual correction of POS tags. Using 'true POS' assignments, instead of the predicted assignments, further improved the performance of the 'hierarchical CHAR-CNN (true POS)' morphology tagger. We denote this hybrid strategy AJATAG and have compared its performance on AJA to MarMoT. We use MarMoT as is, without modifications or adaptations to a hybrid setting, because for the linguists it is an *off-the-shelf* tool that is to be used as is.

Evaluation of the morphological tagging by AJATAG demonstrated favorable performance across multiple evaluation metrics:

- **Field-by-field accuracy** – AJATAG accuracy for the two main *analysis* fields (89.0%, 92.7%, respectively) is higher by up to 7% compared to MarMoT's accuracy (82.3%, 85.6% respectively). It should be noted that the greatest gain in accuracy is in *analysis1*, which of the morphological analysis fields is the richest and most difficult to assign. Both approaches perform well identifying the *enclitic* field with an accuracy greater than 96%.
- **Overall accuracy** – We evaluate the overall accuracy of the morphology taggers using a 'flexible' score, which best mimics real-life usefulness of the tagger as it counts each correct tag separately. The overall accuracy of AJATAG was 91.2%, a little over 2% better than MarMoT (89.0%).
- **Accuracy for words with legal tag combinations** – In TAJA each POS tag has a set of legal values for morphological tags. However, both

[8] https://github.com/technion-cs-nlp/nlp4aja

taggers end up assigning a significant percentage of the words with illegal tag combinations. It is noteworthy, however, that for words that were tagged with legal tag combinations (which are the majority at over 80%) the accuracy of AJATAG went up by 4% to 95.2%, while the accuracy of MarMoT was essentially unchanged.

- **Out of Vocabulary accuracy** – Perhaps the most important predictor for future real-world performance of any tagger is its success with words that are out of vocabulary (OOV), especially as OOV words account for 21% of the TAJA test set. When using predicted POS tags with our hierarchical CHAR-CNN model, the accuracy on the challenging *analysis1* field for OOV words was 57.72%, better than MarMoT by approximately 2% (it also performed better on POS tagging of OOV words with 72.58% vs. MarMoT's 71.35%). However, this important performance indicator is where our hybrid AJATAG strategy delivered its most important fruits. The accuracy of AJATAG in the challenging task of morphologically tagging OOV words is 74.91% and 78.42% for the *analysis1* and *analysis2* fields, respectively, which is significantly better than MarMoT's OOV tagging for these two fields (55.82% and 59.95%, respectively). AJATAG also performs much better in the *additional tags* field for OOV words (85.4% compared to MarMoT's 75.0%).

The justification for the hybrid approach explored herein is in its real-world usefulness, outside of the NLP lab. The 56%–60% accuracy of the off-the-shelf solution for the two most important morphological fields, *analysis1* and *analysis2*, when applied to OOV words is not sufficient for real linguistic work. In contrast, the hybrid AJATag strategy achieved an accuracy level of 74.91%–78.42% on morphological tagging of OOV words, which is expected to be useful for real-world applications, improving upon MarMoT by 18%–19% for this task on both analysis fields. It is reassuring that even without the added human input, our fully automated hierarchical CHAR-CNN performed better than MarMoT on POS and *analysis1* tagging of OOV words. The value of the AJATAG strategy was further confirmed by other performance indicators, including its overall accuracy and its accuracy on words with legal tag combinations, as defined above.

To assess the feasibility of the human interface element in AJATAG, we performed a real-world evaluation of this process. The first-tier POS output was given to two AJA linguists to correct, before moving on to the second-tier morphology tagging. POS tags manually corrected by a senior expert were perceived as the 'true' POS assignment, to which the performance of the automatic taggers as well as the corrections by a junior expert were compared. It is reassuring that both automated taggers, our CHAR-CNN model and MarMoT, performed well at an almost identical accuracy (~89%) relative to the 'true' POS, an accuracy quite similar to the 91% accuracy by the junior expert, who is a PhD candidate with several years of experience in AJA linguistics.

To conclude, while not perfect, the hybrid AJATAG approach provides AJA linguists with a working solution that already impacts their real-world workflow in a way that off-the-shelf tools cannot provide. In the future we plan to continue improving these tools by addressing limitations such as tagging words with illegal tag combinations. Nonetheless, we believe that even in its current form AJATAG could prove useful to linguists as they take on the task of analyzing large untagged AJA corpora. We hope that in the future we will be able to expand the utility of these tools to other Judeo-Arabic dialects.

Acknowledgements

This research was supported by the Israel Science Foundation (grant No. 1191/18). YB was supported by an Azrieli Foundation Early Career Faculty Fellowship.

References

Abdelali, Ahmed, Kareem Darwish, Nadir Durrani, and Hamdy Mubarak. 2016. Farasa: A fast and furious segmenter for Arabic. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 11–16, San Diego, California. Association for Computational Linguistics.

Abdelali, Ahmed, Hamdy Mubarak, Younes Samih, Sabit Hassan, and Kareem Darwish. 2021. QADI: Arabic dialect identification in the wild. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 1–10, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Ahmed, Mohamed AH. 2018. Xml annotation of hebrew elements in judeo-arabic texts. *Journal of Jewish Languages*, 6(2):221–242.

Altaher, Yousef, Ali Fadel, Mazen Alotaibi, Mazen Alyazidi, Mishari Al-Mutairi, Mutlaq Aldhbuiub, Abdulrahman Mosaibah, Abdelrahman Rezk, Abdulrazzaq Alhendi, Mazen Abo Shal, et al. 2022. Masader plus: A new interface for exploring+ 500 arabic nlp datasets. *arXiv preprint arXiv:2208.00932*.

Alyafeai, Zaid, Maraim Masoud, Mustafa Ghaleb, and Maged S. Al-shaibani. 2022. Masader: Metadata sourcing for Arabic text and speech data resources. In

Proceedings of the Thirteenth Language Resources and Evaluation Conference, pages 6340–6351, Marseille, France. European Language Resources Association.

Attia, Mohammed, Younes Samih, Ali Elkahky, Hamdy Mubarak, Ahmed Abdelali, and Kareem Darwish. 2019. POS tagging for improving code-switching identification in Arabic. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 18–29, Florence, Italy. Association for Computational Linguistics.

Ballesteros, Miguel, Chris Dyer, and Noah A. Smith. 2015. Improved transition-based parsing by modeling characters instead of words with LSTMs. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 349–359, Lisbon, Portugal. Association for Computational Linguistics.

Bar-Asher, Moshe. 1992. *La composante hebraïque du judeo-arabe algerien: communautés de Tlemcen et Aïn-Témouchent.* Magnes, Jerusalem.

Belinkov, Yonatan. 2021. Large-scale electronic corpora and the study of middle and mixed Arabic. In *Middle and Mixed Arabic over Time and across Written and Oral Genres: From Legal Documents to Television and Internet through Literature. Proceedings of the IVth AIMA International Conference (Emory University, Atlanta, GA, USA, 12–15 October 2013)*, Publications de l'Institut Orientaliste de Louvain, pages 43–67, Université catholique de Louvain, Louvain-la-Neuve. Peeters.

Bouamor, Houda, Nizar Habash, Mohammad Salameh, Wajdi Zaghouani, Owen Rambow, Dana Abdulrahim, Ossama Obeid, Salam Khalifa, Fadhl Eryani, Alexander Erdmann, et al. 2018. The MADAR Arabic dialect corpus and lexicon. In *LREC*.

Caruana, Rich. 1997. Multitask learning. *Machine Learning*, 28.

Çetinoğlu, Özlem, Sarah Schulz, and Ngoc Thang Vu. 2016. Challenges of computational processing of code-switching. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 1–11, Austin, Texas. Association for Computational Linguistics.

Cohen, Marcel. 1912. *Le parler arabe des Juifs d'Alger*. Collection linguistique, pub. par la Société de linguistique de Paris-4. H. Champion, Paris.

Darwish, Kareem, Mohammed Attia, Hamdy Mubarak, Younes Samih, Ahmed Abdelali, L. Màrquez, M. Eldesouki, and Laura Kallmeyer. 2020. Effective multi-dialectal Arabic POS tagging. *Natural Language Engineering*, 26:677 – 690.

Darwish, Kareem, Hamdy Mubarak, Ahmed Abdelali, Mohamed Eldesouki, Younes Samih, Randah Alharbi, Mohammed Attia, Walid Magdy, and Laura Kallmeyer. 2018. Multi-dialect Arabic POS tagging: A CRF approach. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Dermatas, Evangelos and George Kokkinakis. 1995. Automatic stochastic tagging of natural language texts. *Computational Linguistics*, 21(2):137–163.

Diab, Mona. 2009. Second generation AMIRA tools for Arabic processing: Fast and robust tokenization, POS tagging, and base phrase chunking. In *In 2nd International Conference on Arabic Language Resources and Tools*.

Dos Santos, Cicero and Bianca Zadrozny. 2014. Learning character-level representations for part-of-speech tagging. In *International Conference on Machine Learning*, pages 1818–1826. PMLR.

Duh, Kevin and Katrin Kirchhoff. 2005. POS tagging of dialectal Arabic: A minimally supervised approach. In *Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages*, pages 55–62, Ann Arbor, Michigan. Association for Computational Linguistics.

El-Haj, Mahmoud. 2020. Habibi - a multi dialect multi national Arabic song lyrics corpus. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1318–1326, Marseille, France. European Language Resources Association.

El-Haj, Mahmoud and Rim Koulali. 2013. KALIMAT a multipurpose Arabic corpus. In *Second workshop on Arabic corpus linguistics (WACL-2)*, pages 22–25.

Ferguson, Charles A. 1959. Diglossia. *WORD*, 15(2):325–340.

Habash, Nizar, Mona Diab, and Owen Rambow. 2012. Conventional orthography for dialectal Arabic. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 711–718, Istanbul, Turkey. European Language Resources Association (ELRA).

Habash, Nizar, Ryan Roth, Owen Rambow, Ramy Eskander, and Nadi Tomeh. 2013. Morphological analysis and disambiguation for dialectal Arabic. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 426–432, Atlanta, Georgia. Association for Computational Linguistics.

Habash, Nizar, Abdelhadi Soudi, and Timothy Buckwalter. 2007. *On Arabic Transliteration*, volume 38, chapter 2. Springer Netherlands, Dordrecht.

Hajič, Jan, Otakar Smrž, Petr Zemánek, Petr Pajas, Jan Šnaidauf, Emanuel Beška, Jakub Kracmar, and Kamila Hassanová. 2009. Prague Arabic dependency treebank 1.0. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Hary, Benjamin. 2003. Judeo-Arabic: A diachronic reexamination. *International Journal of The Sociology of Language*, 2003:61–75.

Hochreiter, Sepp and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Inoue, Go, Salam Khalifa, and Nizar Habash. 2022. Morphosyntactic tagging with pre-trained language models for Arabic and its dialects. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1708–1719, Dublin, Ireland. Association for Computational Linguistics.

Inoue, Go, Hiroyuki Shindo, and Yuji Matsumoto. 2017. Joint prediction of morphosyntactic categories for fine-grained Arabic part-of-speech tagging exploiting tag dictionary information. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 421–431, Vancouver, Canada. Association for Computational Linguistics.

Kahn, Lily and Aaron D Rubin. 2017. *Handbook of Jewish Languages: Revised and Updated Edition*. Brill.

Kim, Yoon, Yacine Jernite, David Sontag, and Alexander M. Rush. 2016. Character-aware neural language models. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, page 2741–2749. AAAI Press.

Lafferty, John, Andrew Mccallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289.

Ling, Wang, Chris Dyer, Alan W Black, Isabel Trancoso, Ramón Fermandez, Silvio Amir, Luís Marujo, and Tiago Luís. 2015. Finding function in form: Compositional character models for open vocabulary word representation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1520–1530, Lisbon, Portugal. Association for Computational Linguistics.

Luong, Minh-Thang and Christopher D. Manning. 2016. Achieving open vocabulary neural machine translation with hybrid word-character models. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1054–1063, Berlin, Germany. Association for Computational Linguistics.

Maamouri, Mohamed, Ann Bies, Tim Buckwalter, and Wigdan Mekki. 2004. The Penn Arabic treebank : Building a large-scale annotated Arabic corpus. In *NEMLAR conference on Arabic language resources and tools*, volume 27, pages 466–467. Cairo.

Maamouri, Mohamed, Ann Bies, Seth Kulick, Michael Ciul, Nizar Habash, and Ramy Eskander. 2014. Developing an Egyptian Arabic treebank: Impact of dialectal morphology on annotation and tool development. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2348–2354, Reykjavik, Iceland. European Language Resources Association (ELRA).

McCarthy, John J. 1981. A prosodic theory of non-concatenative morphology. *Linguistic Inquiry*, 12:373–418.

Müller, Thomas, Helmut Schmid, and Hinrich Schütze. 2013. Efficient higher-order CRFs for morphological tagging. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*.

Nivre, Joakim, Željko Agić, Maria Jesus Aranzabe, Masayuki Asahara, Aitziber Atutxa, Miguel Ballesteros, John Bauer, Kepa Bengoetxea, Riyaz Ahmad Bhat, Cristina Bosco, Sam Bowman, Giuseppe G. A. Celano, Miriam Connor, Marie-Catherine de Marneffe, Arantza Diaz de Ilarraza, Kaja Dobrovoljc, Timothy Dozat, Tomaž Erjavec, Richárd Farkas, Jennifer Foster, Daniel Galbraith, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Yoav Goldberg, Berta Gonzales, Bruno Guillaume, Jan Hajič, Dag Haug, Radu Ion, Elena Irimia, Anders Johannsen, Hiroshi Kanayama, Jenna Kanerva, Simon Krek, Veronika Laippala, Alessandro Lenci, Nikola Ljubešić, Teresa Lynn, Christopher Manning, Cătălina Mărănduc, David Mareček, Héctor Martínez Alonso, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Anna Missilä, Verginica Mititelu, Yusuke Miyao, Simonetta Montemagni, Shunsuke Mori, Hanna Nurmi, Petya Osenova, Lilja Øvrelid, Elena Pascual, Marco Passarotti, Cenel-Augusto Perez, Slav Petrov, Jussi Pitulainen, Barbara Plank, Martin Popel, Prokopis Prokopidis, Sampo Pyysalo, Loganathan Ramasamy, Rudolf Rosa, Shadi Saleh, Sebastian Schuster, Wolfgang Seeker, Mojgan Seraji, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Kiril Simov,

Aaron Smith, Jan Štěpánek, Alane Suhr, Zsolt Szántó, Takaaki Tanaka, Reut Tsarfaty, Sumire Uematsu, Larraitz Uria, Viktor Varga, Veronika Vincze, Zdeněk Žabokrtský, Daniel Zeman, and Hanzhi Zhu. 2015. Universal dependencies 1.2. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Owens, J. 2013. *The Oxford Handbook of Arabic Linguistics*. Oxford Handbooks. Oxford University Press.

Pasha, Arfath, Mohamed Al-Badrashiny, Mona Diab, Ahmed El Kholy, Ramy Eskander, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan Roth. 2014. MADAMIRA: A fast, comprehensive tool for morphological analysis and disambiguation of Arabic. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1094–1101, Reykjavik, Iceland. European Language Resources Association (ELRA).

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Plank, Barbara, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 412–418, Berlin, Germany. Association for Computational Linguistics.

Riabi, Arij, Benoît Sagot, and Djamé Seddah. 2021. Can character-based language models improve downstream task performances in low-resource and noisy language scenarios? In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, pages 423–436, Online. Association for Computational Linguistics.

Roth, Ryan, Owen Rambow, Nizar Habash, Mona Diab, and Cynthia Rudin. 2008. Arabic morphological tagging, diacritization, and lemmatization using lexeme models and feature ranking. In *Proceedings of ACL-08: HLT, Short Papers*, pages 117–120, Columbus, Ohio. Association for Computational Linguistics.

Seddah, Djamé, Farah Essaidi, Amal Fethi, Matthieu Futeral, Benjamin Muller, Pedro Javier Ortiz Suárez, Benoît Sagot, and Abhishek Srivastava. 2020. Building a user-generated content North-African Arabizi treebank: Tackling hell. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1139–1150, Online. Association for Computational Linguistics.

Tedghi, Joseph. 2012. 'le livre de Jonas' traduit en judéo-arabe marocain par Samuel Malka: étude linguistique. In *Dynamiques langagières en Arabophonies*, pages 253–290, Zaragoza. Universidad de Zaragoza, Área de Estudios Árabes e Islámicos.

Terner, Ori, Kfir Bar, and Nachum Dershowitz. 2020. Transliteration of Judeo-Arabic texts into Arabic script using recurrent neural networks. In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 85–96, Barcelona, Spain (Online). Association for Computational Linguistics.

Tirosh-Becker, Ofra. 1988. The phonology and topics in the morphology of a Judeo-Arabic translation of the book of Psalms from Constantine (Algeria) / של תרגום לספר תהילים בערבית-יהודית מקונסטנטין (אלג'יריה) פונולוגיה ופרקים במורפולוגיה. Master's thesis, The Hebrew University of Jerusalem.

Tirosh-Becker, Ofra. 1989. On the linguistic uniformity in the "Šarḥ" of the Jews of Constantine / קונסטנטין לשאלת אחדות הלשון בשרח של יהודי. *Proceedings of the World Congress of Jewish Studies /העולמי למדעי היהדות דברי הקונגרס*, .204–197:י

Tirosh-Becker, Ofra. 2011a. Old and new in the translation and commentary of Avot tractate / אבות ופירושה ישן וחדש בתרגום משנת. In *Hikrei ma'arav u-mizrah: studies in language, literature and history presented to Joseph Chetrit (Volume 1) / מוגשים ליוסף שיטרית כרך 1) חקרי מערב ומזרח: לשונות, ספרויות ופרקי תולדה*, Hikrei ma'arav u-mizrah : studies in language, literature and history presented to Joseph Chetrit / ליוסף שיטרית חקרי מערב ומזרח: לשונות, ספרויות ופרקי תולדה מוגשים. Carmel.

Tirosh-Becker, Ofra. 2011b. On dialectal roots in Judeo-Arabic texts from Constantine (east Algeria). *Revue des Études Juives*, 170:227–253.

Tirosh-Becker, Ofra. 2011c. Terms for realia in an Algerian Judeo-Arabic translation of the Hošaʿnot. *Studies in the Culture of North African Jewry*, 1:171–186.

Tirosh-Becker, Ofra. 2012. Mixed linguistic features in a Judeo-Arabic text from Algeria: The Šarḥ to the Hafṭarot from Constantine. In *Language and Nature: Papers presented to John Huehnergard on the Occasion*

of his 60th Birthday, pages 391–406, Chicago. Oxford University Press.

Tirosh-Becker, Ofra. 2014. A reflection of a linguistic reality: An Algerian Judeo-Arabic book for the new year. *Studies in the Culture of North African Jewry*, 3:193–216.

Tirosh-Becker, Ofra and Oren M. Becker. 2022. TAJA corpus: Linguistically tagged written Algerian Judeo-Arabic corpus. *Journal of Jewish Languages*, 10(1):24 – 53.

Wagner, Esther-Miriam and Magdalen Connolly. 2018. Code-switching in judaeo-arabic documents from the cairo geniza. *Multilingua*, 37(1):1–23.

Zaidan, Omar and Chris Callison-Burch. 2014. Arabic dialect identification. *Computational Linguistics*, 40:171–202.

Zalmout, Nasser and Nizar Habash. 2019. Adversarial multitask learning for joint multi-feature and multi-dialect morphological modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1775–1786, Florence, Italy. Association for Computational Linguistics.

Zalmout, Nasser and Nizar Habash. 2020. Joint diacritization, lemmatization, normalization, and fine-grained morphological tagging. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8297–8307, Online. Association for Computational Linguistics.

Zhang, Wen, Yang Feng, Fandong Meng, Di You, and Qun Liu. 2019. Bridging the gap between training and inference for neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4334–4343, Florence, Italy. Association for Computational Linguistics.

Zitouni, Imed. 2014. *Natural Language Processing of Semitic Languages*. Springer.

Zribi, Inès, Mariem Ellouze, Lamia Belguith, and Philippe Blache. 2015. Spoken Tunisian Arabic corpus "STAC": Transcription and annotation. *Research in Computing Science*, 90.

A Data

In this appendix, we detail the transliteration scheme for JA texts used in this paper (Table 9). This table only covers the consonants in JA, as the pronounciation of the vowels in text is not always known.

We also show some of the tag sets used in TAJA. We detail all the POS tags (Table 10), and the morphological tags of the more prominent POS tags (Tables 11, 12, 13 and 14).

Hebrew letter	Transliteration
א	ʾ
ב	b
ג׳	ǧ
ג	ġ
ד	d
ה	h
ו	w
ז	z
ח	ḥ
ט	ṭ
י	y
כ	k
כ׳	x
ל	l
מ	m
נ	n
ס	s
ע	ʿ
פ	f
צ	ṣ
צ׳	ḍ
ק	q
ר	r
ש	š
ת	t

Table 9: Transliteration table for Hebrew (JA) letters.

POS code	Hebrew POS	POS
שע	שם עצם	noun
פע	פועל	verb
מל	מילית	particle
שת	שם תואר	adjective
שפ	שם פרטי	proper noun
מס	מספר	number
כג	כינוי גוף	pronoun
כר	כינוי רמז	demonstrative
כז	כינוי זיקה	relative pronoun
תפ	תואר הפועל	adverb
הצ	הצגה	presentative
תז	תיאור זמן	temporal adjunct
תמ	תיאור מקום	locative adjunct
רת	ראשי תיבות	acronym

Table 10: Legal POS tags in TAJA.

Nouns					
analysis1 code		analysis2 code		additional tags code	
זכר	masculine	יחיד	singular	NA	
נקבה	feminine	רבים	plural		
עברי	Hebrew	זוגי	dual		
ארמי	Aramaic				
לועזי	foreign				

Table 11: Legal morphological analyses for nouns.

Verbs					
analysis1 code		analysis2 code		additional tags code	
(בניין)	(derived stem)	(זמן)	(tense)	(גוף)	(person)
בנ1	I	עב	perfect	י1	1s
בנ2	II	עת	imperfect	יז2	2sm
בנ3	III	צו	imperative	ינ2	2sf
בנ4	IV	בפע	passive participle	יז3	3sm
בנ5	V	בפו	active participle	ינ3	3sf
בנ6	VI	מצ	verbal noun	ר1	1p
בנ7	VII	לפעול	infinitive	רז2	2pm
בנ8	VIII			רנ2	2pf
בנ10	X			רז3	3pm
בנn	passive stem			רנ3	3pf
	related to VII			יחיד	participle sm
בנt	passive stem with a t/tt prefix			יחידה	participle sf
				רבים	participle pm
				רבות	participle pf

Table 12: Legal morphological analyses for verbs.

Adjectives				
analysis1 code		analysis2 code		additional tags code
יחיד	singular masculine	עברי	Hebrew	NA
יחידה	singular feminine	ארמי	Aramaic	
רבים	plural masculine	לועזי	foreign	
רבות	plural feminine			

Table 13: Legal morphological analyses for adjectives.

Proper Nouns				
analysis1 code		analysis2 code		additional tags code
אדם	person	משוערב	Arabized	NA
מקום	place	מתורגם	translated	
עם	people	עברי	Hebrew	
האל	God			

Table 14: Legal morphological analyses for proper nouns.

B Experiments

After determining that the CHAR-CNN model is the best of the three options, we conducted hyperparameter tuning by k-fold cross-validation ($k = 5$). The hyperparameters that we wanted to test are summarized in Table 15, where the reported statistics and standard deviation are over the folds. Rather than report the mean for each hyperparameter test, we report difference between the base configuration result and the hyperparameter result. The value of each hyperparameter in the base configuration appears in parentheses following the name of the hyperparameter. As we are attempting to optimize a large number of hyperparameters, grid search was deemed unfeasible (with a Cartesian product of over 23k hyperparameter combinations). Instead, we test each hyperparameter separately against the base configuration. However, we saw no significant differences between various configurations. This is evident from the table, as in most cases, the results for the tested hyperparameters are within one standard deviation of the base configuration result. Therefore, we continue conducting all our experiments using the original base configuration.

Hyperparameter	value	micro average accuracy (mean)	std	num epochs (mean)
base configuration	**NA**	**0.8908**	**0.0058**	**13**
batch size (8)	4	-0.0023	0.0069	12.6
	16	-0.0034	0.0048	12
directions (2)	1	-0.0004	0.0042	15.2
dropout (0.5)	0.0	-0.0041	0.0036	11.6
	0.3	-0.0048	0.0083	11.8
	0.7	-0.0031	0.0079	13.4
learning rate (0.1)	0.01	-0.0007	0.0050	12.8
	0.05	+0.0009	0.0047	13.4
	0.5	-0.0022	0.0099	12.8
kernel width (6)	4	-0.0046	0.0048	14.2
	8	-0.0053	0.0086	11.2
num kernels (500)	250	-0.0054	0.0049	13.2
	1000	-0.0019	0.0099	12.4
char embedding dim (25)	10	-0.0092	0.0081	15
	50	<-0.0001	0.0045	11.2
word embedding dim (100)	50	-0.0009	0.0047	13.8
	200	+0.0013	0.0027	13.2
hidden dim (100)	50	-0.0009	0.0060	13.6
	200	-0.0006	0.0039	12.6

Table 15: Summary of hyperparameter tuning (base configuration value in parentheses.)

Lexical variation in English language podcasts, editorial media, and social media

Jussi Karlgren, Spotify, Stockholm, Sweden

Abstract The study presented in this paper demonstrates how transcribed podcast material differs with respect to lexical content from other collections of English language data: editorial text, social media, both long form and microblogs, dialogue from movie scripts, and transcribed phone conversations. Most of the recorded differences are as might be expected, reflecting known or assumed difference between spoken and written language, between dialogue and soliloquy, and between scripted formal and unscripted informal language use. Most notably, podcast material, compared to the hitherto typical training sets from editorial media, is characterised by being in the present tense, and with a much higher incidence of pronouns, interjections, and negations. These characteristics are, unsurprisingly, largely shared with social media texts. Where podcast material differs from social media material is in its attitudinal content, with many more amplifiers and much less negative attitude than in blog texts. This variation, besides being of philological interest, has ramifications for computational work. Information access for material which is not primarily topical should be designed to be sensitive to such variation that defines the data set itself and discriminates items within it. In general, training sets for language models are a non-trivial parameter which are likely to show effects both expected and unexpected when applied to data from other sources and the characteristics and provenance of data used to train a model should be listed on the label as a minimal form of downstream consumer protection.

1 Genres and podcast transcripts

The way human language is used varies across channels and styles, and we have for the longest while made a clear distinction between spoken and written language as two major distinctive modes of communication (Cederschiöld, 1897; Ong, 1982; Biber, 1991; Coulmas, 2003).

The differences between writing and reading can to a large extent be related to situational differences: where speech has been used in transient situations in which interlocutors are present, writing has typically been used in asynchronous communication with participants at a remove from each other. This distinction has through the introduction of communication technologies become less and less clear-cut. Written language is used for momentary and fleeting conversations with little planning or editorial oversight; spoken language material is created, published and distributed in ways which are more formal and more permanent and archival than before.

Podcasts are a new medium and a new format for spoken language. The styles of language use in podcasts are as yet unformed and have not yet coalesced into stable functional and generally accepted genres: podcast material will require us to recalibrate many of the assumptions we make about how language is used. Recently, a collection of over 100,000 podcast episodes, including automatically generated transcripts, has been released for the purposes of retrieval and summarisation experimentation. The companion paper released with the podcast material set gives some indicative differences between the transcripts and written language as represented by the Brown corpus (Francis and Kucera, 1967) and shows i.a. that the frequency of amplifiers and personal pronouns is greater than in the various genres represented in the Brown corpus (Clifton et al., 2020).

This paper demonstrates how some such differences across text collections of different types are indicative of genre differences, some of which can be expected to depend on how spoken genres continue to evolve with changing technology and evolving situations of usage. This examines differences in anchoring, subjective language, and discourse handling, which can all be expected to be dimensions in which podcast language will differ from written genres.

Podcasts are a rapidly evolving medium. The variation and volatility is great and we can expect that only a few years from now there will be new formats of language use not represented in the present collection. These measurements are intended to inspire the systematic exploration of such differences as they occur, to make possible documentation of current and future

changes in the medium, to make explicit differences that may have effects on the applicability of language models trained on one type of material on another, and to ensure that application to classification, retrieval, or large scale extraction of information is informed and sensitive to those systematic differences that might impact results.

2 Data Overview

Seven data sets were used for these experiments. These data sets are of varying age and collected with various methods, but have all been used in research and benchmarking projects and are selected by virtue of being accessible for experimentation and further study. The representativeness of the corpora may vary: movie scripts change over time as the craft of writing and acting evolves; conventions in phone conversations change as new technologies cater to new use cases; social media platforms, with various conventions and various technological affordances, go in and out of fashion; editorial media shift their focus and their offerings according to the shifting constitution and preferences of their audiences. The editorial media data set is the one most clearly governed by conventions and constraints imposed by audience expectations for the genre and is likely to be the data set with least change over time. These changes are all likely to affect the stylistic statistics on reported below in various ways; the differences found between genres are robust enough to majorise the within genre differences over time.

Editorial media A collection of Associated Press newswire text from year 1989-1990 made available for experimentation in various shared tasks as part of the TIPSTER corpus (Harman and Liberman, 1993). These represent edited text conforming to standard written English language usage.

> *... Citing financial disarray in Massachusetts government, a major bond rating agency cut the state's credit rating Friday for the second time this year, a move that could add millions to borrowing costs. The decision by Standard & Poor's Corp. to downgrade Massachusetts bonds from AA-to A represents a harsh assessment of the fiscal policies of Gov. Michael S. Dukakis and the state Legislature. "The state's economy remains strong, while debt and fiscal management display serious weaknesses," the agency said. ...*

Social media Data from the Blog Authorship Corpus which consists of a large age- and gender-balanced collection of about 700 000 English language blog posts collected for the purpose of experimentation with authorship attribution (Schler et al., 2006). These are intended to represent informal written language in a variety of subgenres.

> *... Yesterday I learned a new programming language, Groovy . Well, I wrote a simple program in Groovy. I need to do much more with it before I learn to "think in Groovy." This is important. There's a huge benefit to learning a new programming language, so much so that The Programatic Programmers recommend learning a new language every year. Learning a new programming language can be difficult. Let's be precise: learning to write working programs in a new language is relatively easy, but the first impulse is to think in the style of the languages you already know and write programs using the syntax of the new language. ...*

Microblogs A set of mostly English language microblog posts from Twitter collected for analysis of public opinion during the fall of 2017. [1]. These are intended to represent real-time language use, but in fact contain a large number of press release announcements, news headlines, and links to further reading.

- *Mystery Fanged Sea Creature Washes Up on Texas Beach after Hurricane Harvey* `URL`
- *Hope and kudos for hurricane victims in healthcare:* `URL`
- *as i sit in this heat i also wish tha best for those that caught harvey cause i know theyre worse off n im grateful we ain get hit directly*
- *Having a gun license is what you're thinking about after a disaster? If you're in Taaaxas. #Harvey* `URL`
- `@UId` *Hi, sorry missed your question! 7-8pm at harvey hadden, this will be the only one i'm afraid*

Podcast transcripts A large collection of automatically generated English language podcast transcripts released by Spotify for research purposes, with episodes representing a variety of podcast formats, styles, levels of formality, and topics (Clifton et al., 2020). The transcripts include sentence breaks automatically inferred by the transcription system.

> *... Only on my hands no with my hips ever. So first what I did was visiting a doctor because every time when I was trying to stretch*

[1] The post ids are available at `http://www.lingvi.st/corpora/storm.txt`

myself like to take stretch classes, I ended up with like a really bad pain for like a few weeks or months. So then they visited doctor and I really like he told me that my spine like ...

Movie scripts A collection of English language movie scripts from the Film Corpus (Walker et al., 2012). The corpus has separated dialogue from scene descriptions and director instructions; for the purpose of this study, only the dialogue portion has been used, as a sample of language which is produced in written form but intended to represent natural speech.

...
- What's that shit?
- A book. It's called reading. You should try it some time.
- You wanna read something. Read between the lines.
- Well here's something even you can relate to. Albert got a lotta trim.
- That genius thing is a babe magnet.
- Lemme see that book.
...

Telephone conversations The Switchboard corpus is a collection of transcribed English language telephone conversations on a variety of topics (Godfrey et al., 1992; Godfrey and Holliman, 1997). For this study a separately annotated portion which is freely available is used (Jurafsky et al., 1997). This is intended to represent the character of spontaneous unscripted speech. This transcription is fairly carefully done to preserve e.g. interruptions and overlapping speech, in contrast with the podcast transcription.

...
— What kind of ...
— Okay.
— ... eating out do you enjoy?
— Well, I like dining out.
— Of course, it means that I don't have to cook.
— Right .
— But, um, I'm a divorced woman.
— I have one child ...
— Uh-huh.
— ... and, you know, when, when we dine out we go to like medium priced restaurants.
— Uh—huh.
— I don't, I don't particularly
— I think it's sort of a waste of money to go real, to a real high priced restaurant.
— Do you go like home cooking, like Black-Eyed Pea and that kind of thing or ...
— Um, e-, n-
— ... cafeteria?
— Not really.
— We go wh-, more for the, uh, Chinese ...
— Me too .
— ... and Italian ...
— Uh-huh.
— ... and stuff like that. Mexican, stuff ...
— Mexican,
— uh-huh.
— ... that I can't cook .
— Uh, we do too.
— We do the same.
— Yeah.
...

Popular lectures The popular science TED talk series on "technology, entertainment, and design" provide transcripts of lectures given by the speakers. The lectures are information-dense, but informal and entertaining in style and are mostly monologues, with the occasional conversational interview. A selection of such transcripts has been made available for experimentation (Banik, 2017).

I'd like to tell you the tale of one of my favorite projects. I think it's one of the most exciting that I'm working on, but I think it's also the simplest. It's a project that has the potential to make a huge impact around the world. It addresses one of the biggest health issues on the planet, the number one cause of death in children under five. Which is...? Water-borne diseases? Diarrhea? Malnutrition? No. It's breathing the smoke from indoor cooking fires — acute respiratory infections caused by this.

100 000 sentences from each source were sampled for inclusion in this study, using the Natural Language Toolkit (NLTK) for sentence segmentation which splits the text to sentences at major delimiters (".", "!", "?") and at paragraph breaks (Bird, 2006). Some quantitative data for the samples are given in Table 1. Noticeable is that the sentence length varies considerably across the collections. This reflects both genre variation and transcription practice, as can be seen in the above example extracts: the movie scripts contain very short sentences authored to describe rapid dialogue and overlapping turns, the phone conversation transcripts render short turns and interruptions as separate sentences, where, by contrast, the podcast transcripts have longer turns on average. Repeated samples were drawn to ensure stability of the measures made, and all measures and statistics given in the following tables are averaged

across a number of resamplings, rounded to two significant figures.

3 Dimensions of variation

The measures examined in this study focus on readily inspectable aspects of language use where spoken language and informal channels traditionally are assumed to show difference to written formal genres. Spoken language due to its immediacy and synchronous nature frequently has more overt markers for interpersonal functions, and utilises different textual functions to organise the discourse. Since written genres more frequently are used for abstract and complex topical matter, it is to be expected that those ideational functions that concern argumentation and logical structure are rendered differently. Biber and colleagues, in their studies on register variation across several languages (Biber, 1995), posit a number of variational dimensions using factorial analyses and then formulate a low dimensional space of *functional bases* in which they position the genre samples such as lectures, face-to-face conversations, broadcasts, private letters, academic prose, offical documents, and many more.

This study uses a subset of the variables examined by Biber and colleagues (the variables used by Biber are variously accessible for automated analysis). The addition of podcast material to the data used by Biber are likely to extend the variational dimensions posited by his original study, since podcast material cuts across many of the suggested dimensions such as "involved vs informational" which separates e.g. speeches from e.g. academic prose; "narrative vs non-narrative", which separates fiction text from e.g. face-to-face interaction; "textual vs situational reference", which separates e.g. phone conversations from official documents and so forth. Podcasts incorporate material with the situatedness of personal conversations to the abstraction of formal lectures, and material with the immediacy and interactive online planning of live dialogue to the editorially oriented production qualities of broadcast news. We can expect that many of the variational dimensions are relevant for podcasts even as new conventions and new genres gradually develop.

Spoken unscripted language is characterised by explicit features related to the organisation of discourse which involve turn-taking, interruptions, dysfluencies, and repair. These are somewhat challenging to study with the given collections, especially as transcription oftentimes removes and normalises much of the signal. Notably, in the present collections, while the phone conversation transcripts render turn-taking in detail, the podcast transcriptions leave out overlapping speech.

This study focusses on features of language use which is *situated*, where the participants are synchronously present during the communicative situation as opposed to communication where the author or speaker is separated from the audience, and *personal and subjective*, where the attitude and stance of the author or speaker is clearly expressed and modulated to capture the attention and fit the reactions of the intended audience, in contrast to language framed to be formal and couched in objective terms and expressions, abstracted from the present situation.

The surface features to be expected are more attitudinal and overtly subjective language, with intensifiers, first and second person pronouns, more present tense and narratives, more questions and affirmations than in scripted and planned language use.

4 Attitude and Affect in Language

Subjective language is of interest for many reasons, but not least for its potential applications in information retrieval and text categorisation. Since the introduction of computational sentiment analysis as a research topic (Qu et al., 2004) various efforts to extend or typologise the field have been explored, (Karlgren et al., 2004; Karlgren, 2009; Feldman, 2013; Ravi and Ravi, 2015) and many mostly lexical approaches were implemented for commercial application. Now, with computational methods that allow full scope over an entire utterance without relying on single items, some of the lexical approaches are less immediately impressive than before, but for reasons of transparency, many are still in use in practical applications and they correlate well with findings from non-lexical approaches. For the present experiments, a standard lexicon of polar items has been used to represent the manifold expressions of human emotion found in text (Hu and Liu, 2004), and the incidence of items from the lexicon are shown in Table 2[2].

The table gives counts both per word, i.e. how many of the tokens of the collection sample were polar evaluative lexical items (left half of the table), and per sentence, i.e. how many sentences of the 100 000 sample contained a polar evaluative lexical item (right half of the table).

The results show that podcast transcripts have a noticeably higher incidence of positive polarity items and lower incidence of negative polarity items than written genres and that popular lectures exhibit much the same distribution. News stories exhibit more negative polarity than positive polarity items, which is likely to

[2]The item "*like*" was removed from the list of positive items, since it is very frequent in the spoken language material as a non-attitudinal discourse particle.

Table 1: Descriptive statistics for the seven language collections, comparing average sentence length.

	Editorial media	Social media	Microblogs	Podcast transcripts	Movie scripts	Phone calls	Popular lectures
Number of sentences	100 000	100 000	100 000	100 000	100 000	100 000	100 000
Number of words	2 200 000	1 700 000	1 800 000	1 700 000	720 000	720 000	1 600 000
Words per sentence	22	17	18	17	7.2	7.2	16
Year of publication	1989-1990	2004	2017	2019	before 2010	early 1990s	2017

have to do with editorial considerations: negative news drive reporting. This probably explains a similar imbalance for the microblog posts, which to a large extent are commentary to current news stories. The two more traditional spoken genres have much lower counts of both polarities, which may mean that the lexicon used here is not optimised for spoken material or that spoken language demonstrates polarity more often in constructional items rather than purely lexical ones ("*This put me off.*").

5 Amplification

Amplifiers are linguistic items that serve to increase the perceived strength of an evaluative expression. They are typically constructed as adverbials, as shown in Example (1) (Quirk et al., 1985, §7.57 a) and in this study such items are used, and other amplifying constructions are left aside. Amplifiers can be subcategorised in several ways, and here a three-way distinction is made. *Gradation* amplifiers increase the intensity of a gradal expression: (*very*, *immensely*, *substantially*, *fucking*); *affirmation* amplifiers emphasise the commitment of the speaker to the sentiment (*truly*, *really*); and *surprise* amplifiers communicate that the qualities under consideration are unexpected or anomalous (*amazingly*, *surprisingly*, *unusually*). These distinctions are of course not independent of each other. The amplifiers used in this study are given in Appendix A.

(1) a. Hurricane Irma is a ***very*** dangerous storm. (MICROBLOG)
b. The ***immensely*** popular "Star Wars" isn't much good for teaching science. (NEWS)
c. It just ***fucking*** cool. (PODCAST)
d. If you're ready to find out who you are deep down and live a ***truly*** authentic life. (PODCAST)
e. Now if you use the right kind of atoms and you get them cold enough, something ***truly*** bizarre happens. (LECTURES)
f. My husband is he's ***really*** sweet. (PODCAST)
g. Leaders of corporate America say business is ***surprisingly*** good. (NEWS)
h. That was interesting, and ***surprisingly*** nice. (BLOG)

The incidence of amplifiers in the seven collections are given in Table 3. We find that the podcast material has an order of magnitude higher number of amplifiers than most other genres. Popular lectures also exhibit a similarly high incidence of amplifiers, but there is a difference in how they are distributed over the subcategories: podcasts show a very high incidence of *affirmation* amplifiers, which take purchase in the presence of the speaker in the communicative situation. This is one of the most differentiating features between podcasts and popular lectures, which otherwise exhibit many similar characteristics.

6 Negation

Negation is a foundational semantic operator whose exact semantic function on the meaning of an utterance can be discussed and modelled at length (Von Klopp, 1993, e.g.). Negation can affect an entire clause ("*I didn't eat the cookies.*") or more locally, a constituent of a clause ("*I will eat no more cookies.*"). In English, clausal negation most often is formed through the negative verbal affix "*n't*", which in written or more formal registers, or when emphasised, often is rendered as the separate lexical item "not". Local negation is formed through prefixing the negated component with "*no*" or "*not*", or by using more elaborate construction such as "*neither ... nor*", "*nobody*", "*none*", or "*never*" (Quirk et al., 1985, §10.55ff). Negation has an obvious relation to polarity and antonymy which has motivated great interest in research on methods for the practical handling of negation in sentiment analysis and related experiments and applications (Choi and Cardie, 2009; Tanushi et al., 2013; Mohammad et al., 2013; Kiritchenko et al., 2014; Reitan et al., 2015, i.a.). Some examples of negation and its effect on polarity are given in Example (2). In this study, negation is included as an example of an accessible semantic operator useful for modulation and modification of attitudinal expressions. The list of negations used in this study, compiled from Quirk et al. (1985) and Biber (1995) is given in Appendix B and the incidence of negations is given in Table 4. We can here observe how informal genres, unsurprisingly, exhibit many more contracted forms than the written material. We also find that the incidence of

Table 2: Occurrence and proportion of negative and positive polar lexical items from Hu and Liu (2004) in seven collections of language, per word and per sentence in a sample of 100 000 sentences from each collection.

	Per word				Per sentence	
	Positive		Negative		Positive	Negative
Editorial media	39 000	(1.8 %)	56 000	(2.6 %)	30 000	39 000
Social media	44 000	(2.6 %)	41 000	(2.5 %)	31 000	28 000
Microblogs	27 000	(1.5 %)	42 000	(2.3 %)	21 000	29 000
Podcast transcripts	46 000	(2.7 %)	29 000	(1.7 %)	33 000	21 000
Movie scripts	16 000	(2.2 %)	18 000	(2.6 %)	14 000	16 000
Phone conversations	16 000	(2.3 %)	9 000	(1.3 %)	15 000	8 100
Popular lectures	41 000	(2.6 %)	29 000	(1.8 %)	31 000	22 000

Table 3: Occurrence and proportion of lexical amplifiers (listed in Appendix A) in seven collections of language, per word and per sentence in a sample of 100 000 sentences from each collection.

	Editorial media	Social media	Microblogs	Podcast transcripts	Movie scripts	Phone conversations	Popular lectures
	Per word						
amplifiers	3 400	8 100	1 800	13 000	2 400	5 200	11 500
gradation	2 100	3 200	840	4 400	1 400	1 500	5 800
	(0.097 %)	(0.19 %)	(0.046 %)	(0.26 %)	(0.20 %)	(0.20 %)	(0.37 %)
affirmation	710	4 200	480	7 300	800	3 500	4 100
	(0.033 %)	(0.26 %)	(0.026 %)	(0.43 %)	(0.11 %)	(0.49 %)	(0.26 %)
surprise	580	680	470	950	160	220	1 600
	(0.027 %)	(0.041 %)	(0.026 %)	(0.055 %)	(0.021 %)	(0.031 %)	(0.10 %)
	Per sentence						
gradation	2 000	3 000	820	3 900	1 400	1 300	5 100
affirmation	700	3 900	440	6 400	780	3 400	4 100
surprise	570	660	410	910	160	220	1 600

negation in general is higher in social media and podcasts than in the other material. There are many hypothetical explanations for this observation which need further study: a tentative explanation is that negation is at times used as a discourse marker ("*No, no, no, no, we can't do that.*" or even "*No, you are right.*")

(2) a. We would continue to pursue the accelerator technology, but at the moment it is ***not*** as mature as fission reactors. (NEWS)
b. And it's crazy how it's it's ***not*** crazy. (PODCAST)
c. My boat got hit by #IrmaHurricane the ranch is #flooding from #irma but #hankjr02 is following me on Twitter, so it ***can't*** be all bad. (MICROBLOG)
d. and, it's ***not*** very expensive that way. (PHONE)
e. And that is ***not*** bad at all. (PHONE)
f. Ladies and gentlemen, a picture is ***not*** worth a thousand words. In fact, we found some pictures that are worth 500 billion words. (LECTURES)

7 Interrogatives

The incidence of interrogative utterances, defined as sentences that end with a "?", differs across the collections as shown in Table 5. It is likely, here as in preceding statistics, that the results are influenced by conventions for transcription which vary across the spoken genres, but the podcast material which is the only automatically transcribed material shows a higher incidence of questions than some of the other genres, rather than the lower incidence which might be expected from transcription errors. The movie script collection stands out here, with every sixth sentence a question, reflecting the type of conversational to-and-fro characteristic of the genre.

8 Situatedness

Personal pronouns are used when the author or speaker and the audience have a shared understanding of the context they are in. First and second person pronouns are less prevalent in formal discourse and more prevalent in face-to-face conversation than in other situations; narrative discourse will show a higher propor-

Table 4: Occurrence and proportion of negated sentences in seven collections of language in a sample of 100 000 sentences from each collection(negations used are listed in Appendix B).

	Editorial media	Social media	Microblogs	Podcast transcripts	Movie scripts	Phone conversations	Popular lectures
negations	17 000	24 000	6 800	28 000	1 900	12 000	17 000
"no", "not"	10 000	9 000	3 500	9 600	4 200	3 800	8 400
contractions	3 800	9 900	4 000	13 000	8 800	7 100	7 300
constructions	1 800	2 500	1 300	2 100	2 300	1 100	1 300

Table 5: Occurrence and proportion of interrogative sentences in seven collections of language in a sample of 100 000 sentences from each collection.

	Editorial media	Social media	Microblogs	Podcast transcripts	Movie scripts	Phone conversations	Popular lectures
Questions	730	7 100	2 800	7 600	17 000	3 700	8 000

tion of third person pronouns than non-narrative discourse. The expected differences are found in the data, as shown in Table 6. These counts include reflexives ("*myself*") and possessives ("*our*", "*ours*"); the third person counts do not include "it"; the second person counts include impersonal "you" as in "when you bake bread you usually add some kind of leavening". Notable is firstly (and unsurprisingly) the high incidence of personal pronouns in the spoken genres together with the social media texts compared to the two other genres. Secondly, notable is the large number of second person pronouns in podcast material, movie scripts, and phone conversations, reflecting the dyadic conversational format in many of them. Thirdly, the large number of 1st person plural pronouns in the popular lecture data, reflecting the genre-specific pattern of including the audience in an utterance ("*When we think about why we hear, we don't often think about the ability to hear an alarm or a siren, although clearly that's an important thing.*") . A final striking observation is the consistently low level of reference to feminine correlates across all collections.

Another measure of situatedness is the distribution of lexical categories over content words. Table 7 shows how verbs are less common and proper nouns are more common in editorial media and in microblogs compared to the other four genres. These counts are based on part of speech tagging as provided by the NLTK part of speech tagger (Bird, 2006). The difference is most likely related to news reporting being based on participating people, organisations, and locales. By contrast, the relative occurrence of verbs is higher in the spoken genres and in social media.

Shared across all genres except the news material is the preponderance of present tense in comparison with past tense as shown in Table 8. This is an indicator of narrative discourse, where language is used to describe something that preceded the communicative situation. The news genre is highly focussed on reporting past events and this is reflected in the tense representation. These counts are also based on the NLTK part of speech tagger, which provides separate labels for past tense verbal forms. Some sentences have mixed tense, subclauses with a tense different from the matrix clause, e.g., or other more complex verbal structure and are omitted from the table.

9 Concluding Observations

This initial study demonstrates some clear differences in lexical content between transcribed podcast material and other collections of language data: editorial text, social media, both long form and microblogs, dialogue from movie scripts, transcribed phone conversations, and popular lectures. Most of the recorded differences are as might be expected, reflecting known or assumed difference between spoken and written language, between dialogue and soliloquy, and between scripted formal and unscripted informal language use. Most notably, podcast material, compared to the hitherto typical training sets from editorial media, is characterised by being in the present tense, and with a much higher incidence of pronouns and negations. These characteristics are, unsurprisingly, largely shared with social media texts. Where podcast material differs from social media material is in its attitudinal content, with many more amplifiers and much less negative attitude than in blog texts. There is a solid base to explain these differences in the studies by Biber referred to above, and in the more general notion of metafunctions of language which are utilised with various relative strength across communicative situations.

It is to be expected that the results presented in this study will age rapidly with respect to their details: the podcast medium will evolve and new genres and stylistic conventions will emerge an coalesce in the near future as podcasts gain a broader audience, more creators, and further situations of use. The popular lec-

Table 6: Occurrence of personal pronouns and their proportion of the vocabulary in seven collections of language.

	Editorial media	Social media	Microblogs	Podcast transcripts	Movie scripts	Phone conversations	Popular lectures
1 p sg	8 000	96 000	6 000	77 000	33 000	36 000	40 000
	(0.37 %)	(5.8 %)	(0.33 %)	(4.5 %)	(4.6 %)	(5.0 %)	(2.5 %)
2 p	2 500	16 000	7 200	52 000	23 000	18 000	30 000
	(0.11 %)	(0.96 %)	(0.39 %)	(3.0 %)	(3.2 %)	(2.6 %)	(1.9 %)
3 p sg m	22 000	14 000	4 700	14 000	9 000	3 100	7 600
	(1.0 %)	(0.82 %)	(0.26 %)	(0.81 %)	(1.2 %)	(0.44 %)	(0.48 %)
3 p sg f	4 500	8 500	2 000	5 900	4 100	2 000	3 700
	(0.21 %)	(0.51 %)	(0.11 %)	(0.34 %)	(0.56 %)	(0.29 %)	(0.23 %)
1 p pl	5 500	12 000	7 000	18 000	5 500	8 000	32 000
	(0.25 %)	(0.74 %)	(0.38 %)	(1.0 %)	(0.76 %)	(1.1 %)	(2.0 %)
3 p pl	11 000	7 000	4 800	13 000	2 600	9 200	15 000
	(0.50 %)	(0.42 %)	(0.26 %)	(0.74 %)	(0.36 %)	(1.3 %)	(0.97 %)

Table 7: Distribution of lexical categories for content words and their proportion of the vocabulary in seven collections of language based on NLTK part of speech tagging.

	Editorial media	Social media	Microblogs	Podcast transcripts	Movie scripts	Phone conversations	Popular lectures
verbs	350 000	320 000	230 000	370 000	150 000	140 000	300 000
	(16 %)	(19 %)	(12 %)	(21 %)	(21 %)	(20 %)	(19 %)
nouns	470 000	310 000	400 000	250 000	110 000	99 000	220 000
	(22 %)	(19 %)	(22 %)	(14 %)	(15 %)	(14 %)	(13 %)
proper nouns	260 000	92 000	460 000	63 000	52 000	21 000	51 000
	(12 %)	(5.5 %)	(25 %)	(3.7 %)	(7.2 %)	(2.9 %)	(3.1 %)
adjectives	160 000	120 000	170 000	96 000	36 000	39 000	110 000
	(7.4 %)	(7.1 %)	(9.2 %)	(5.6 %)	(5.0 %)	(5.4 %)	(6.7 %)

Table 8: Verb tense of sentences in seven collections of language in a sample of 100 000 sentences from each collection. Sentences with mixed tense or complex verb chains omitted.

	Editorial media	Social media	Microblogs	Podcast transcripts	Movie scripts	Phone conversations	Popular lectures
present tense	14 000	32 000	25 000	41 000	39 000	35 000	45 000
past tense	54 000	26 000	25 000	14 000	13 000	11 000	21 000

tures, an offshoot from classical academic lectures, but with their form modified by new transmission channels and by influence from other staged presentations, shows one direction of development which is clearly related to podcasts; we should expect some podcasts to adhere to this genre, while others will be more like drama and scripted speech, and some continue to exhibit similarities to more unscripted and informal conversation. Across all genres, the difference between *he* and *she* in their various forms was dramatic — this is something that may change over time.

These observations have some direct ramifications for computational work. Firstly, any useful approach to information access for material which is not primarily topical should be designed to be sensitive to such variation that defines the data set itself and discriminates items within it. More generally, training sets for language models are a non-trivial parameter which are likely to show effects both expected and unexpected when applied to data from other sources. The characteristics and provenance of data used to train a model should be listed on the label as a minimal form of downstream consumer protection. What these counts specifically demonstrate is that filtering the a data set through application of "stoplists" or other feature reduction methods or assessing the quality of language models using gold standards built on referential semantics based on nouns (cf. Karlgren (2019)) will reduce the richness of expression more in pronoun-rich and verb-rich genres than in those with less pronouns and verbs.

The variation demonstrated by the lexical tables given here is of obvious philological interest, casting light on how human communicative behaviour is modulated by the channel over which it proceeds. These reported statistics are but a scratch on the surface: more sophisticated and hypothesis-driven methods will be able to present more unified underlying variables and models with more explanatory power.

Acknowledgments

The author wishes to thank his colleagues Rosie Jones and Sravana Reddy for insightful comments which have contributed greatly to the quality of the paper.

References

Banik, Rounak. 2017. TED Talks: Data about TED Talks. *Dataset on Kaggle*, Version 3.

Biber, Douglas. 1991. *Variation across speech and writing*. Cambridge University Press, Cambridge.

Biber, Douglas. 1995. *Dimensions of register variation: A cross-linguistic comparison*. Cambridge University Press.

Bird, Steven. 2006. NLTK: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72. International Committee for Computational Linguistics.

du Bois, John W., Wallace L. Chafe, Charles Meyer, Sandra A. Thompson, Robert Englebretson, and Nii Martey. 2000-2005. *Santa Barbara corpus of spoken American English*. Linguistic Data Consortium, Philadelphia.

Cederschiöld, Gustaf. 1897. *Om svenskan som skriftspråk [Swedish as a written language]*. Wettergren & Kerber, Gothenburg.

Choi, Yejin and Claire Cardie. 2009. Adapting a polarity lexicon using integer linear programming for domain-specific sentiment classification. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 590–598. Association for Computational Linguistics.

Clifton, Ann, Sravana Reddy, Yongze Yu, Aasish Pappu, Rezvaneh Rezapour, Hamed Bonab, Maria Eskevich, Gareth Jones, Jussi Karlgren, Ben Carterette, et al. 2020. 100,000 podcasts: A spoken english document corpus. In *Proceedings of the 28th International Conference on Computational Linguistics (Coling)*. International Committee for Computational Linguistics.

Coulmas, Florian. 2003. *Writing systems: An introduction to their linguistic analysis*. Cambridge University Press.

Feldman, Ronen. 2013. Techniques and applications for sentiment analysis. *Communications of the ACM*, 56(4):82–89.

Francis, W Nelson and Henry Kucera. 1967. *Computational analysis of present-day American English*. Brown University Press, Providence.

Godfrey, John J and Edward Holliman. 1997. Switchboard-1 Release 2. *Linguistic Data Consortium, Philadelphia*, 926.

Godfrey, John J, Edward C Holliman, and Jane McDaniel. 1992. SWITCHBOARD: Telephone speech corpus for research and development. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 1, pages 517–520. IEEE.

Harman, Donna and Mark Liberman. 1993. *TIPSTER Complete LDC93T3A Web Download*. Linguistic Data Consortium, Philadelphia.

Hu, Minqing and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the 10th ACM SIGKDD International conference on Knowledge discovery and data mining (KDD)*, pages 168–177. Association for Computing Machinery.

Jurafsky, Dan, Elizabeth Shriberg, and Debra Biasca. 1997. Switchboard SWBD-DAMSL labeling project coder's manual, Draft 13. *Boulder Institute of Cognitive Science Technical Report*.

Karlgren, Jussi. 2009. Affect, appeal, and sentiment as factors influencing interaction with multimedia information. In *Theseus ImageCLEF workshop on visual information retrieval evaluation*, pages 8–11.

Karlgren, Jussi. 2019. How lexical gold standards have effects on the usefulness of text analysis tools for digital scholarship. In *International Conference of the Cross-Language Evaluation Forum for European Languages (CLEF)*, pages 178–184. The CLEF Initiative.

Karlgren, Jussi, Gunnar Eriksson, Stefano Mizzaro, Paul Clough, Mark Sanderson, Kristofer Franzén, and Preben Hansen. 2004. Reading Between the Lines: Attitudinal expressions in text. In *Proceedings of the AAAI Spring Symposium Workshop on Exploring Attitude and Affect in Text: Theories and Applications*. American Association for Artificial Intelligence.

Kiritchenko, Svetlana, Xiaodan Zhu, and Saif M Mohammad. 2014. Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research*, pages 723–762.

Mohammad, Saif M, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. In *Second Joint Conference on Lexical and Computational Semantics (* SEM)*, pages 321–327.

Ong, Walter J. 1982. *Orality and literacy*. Routledge.

Qu, Yan, James Shanahan, and Janyce Wiebe. 2004. *Proceedings of the AAAI Spring Symposium Workshop on Exploring Attitude and Affect in Text: Theories and Applications*. American Association for Artificial Intelligence.

Quirk, Randolph, Sidney Greenbaum, Geoffrey Leech, and Jan Svartvik. 1985. *A comprehensive grammar of the English language*. Longman, London.

Ravi, Kumar and Vadlamani Ravi. 2015. A survey on opinion mining and sentiment analysis: tasks, approaches and applications. *Knowledge-Based Systems*, 89:14–46.

Reitan, Johan, Jørgen Faret, Björn Gambäck, and Lars Bungum. 2015. Negation scope detection for twitter sentiment analysis. In *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 99–108.

Schler, Jonathan, Moshe Koppel, Shlomo Argamon, and James W Pennebaker. 2006. Effects of age and gender on blogging. In *Proceedings of the AAAI Spring Symposium Workshop on Computational approaches to analyzing weblogs*, pages 199–205. American Association for Artificial Intelligence.

Tanushi, Hideyuki, Hercules Dalianis, Martin Duneld, Maria Kvist, Maria Skeppstedt, and Sumithra Velupillai. 2013. Negation scope delimitation in clinical text using three approaches: NegEx, PyConTextNLP, and SynNeg. In *19th Nordic Conference of Computational Linguistics (NODALIDA)*, pages 387–474. Linköping University Electronic Press.

Von Klopp, Ana. 1993. *Negation: Implications for theories of natural language*. Ph.D. thesis, University of Edinburgh.

Walker, Marilyn A, Grace I Lin, and Jennifer Sawyer. 2012. An Annotated Corpus of Film Dialogue for Learning and Characterizing Character Style. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)*, pages 1373–1378. European Language Resources Association (ELRA).

A Amplifiers

gradation amplifiers	affirmation amplifiers	surprise amplifiers
very	absolutely	amazingly
awfully	definitely	dramatically
completely	famously	drastically
enormously	genuinely	emphatically
entirely	immaculately	exceptionally
exceedingly	overly	extraordinarily
excessively	perfectly	fantastically
extremely	really	horribly
fucking	severely	incredibly
fuckin	surely	insanely
greatly	thoroughly	phenomenally
highly	truly	remarkably
hugely	undoubtedly	ridiculously
immensely		strikingly
intensely		surprisingly
particularly		terribly
radically		unusually
significantly		wildly
strongly		wonderfully
substantially		amazing
totally		dramatic
utterly		drastic
vastly		emphatic
		exceptional
		extraordinary
		fantastic
		incredible
		phenomenal
		remarkable
		striking
		surprising
		unusual

B Negations

Negations are taken from Quirk et al. (1985, §10.54ff) and Biber (1995).

analytic	no, not
contractions	n't, ain't, aren't, arent, aren't, can't, cannot, cant, couldn't,, didn't, doesn't, don't, hadn't, hasn't, haven't, isn't, mightn't, mustn't, shouldn't, wasn't,, weren't, won't, wouldn't
constructional	neither, never, nor, none, nobody, no-one, without, sans, w/o

Contextualized language models for semantic change detection: lessons learned

Andrey Kutuzov, University of Oslo, Norway andreku@ifi.uio.no
Erik Velldal, University of Oslo, Norway erikve@ifi.uio.no
Lilja Øvrelid, University of Oslo, Norway liljao@ifi.uio.no

Abstract We present a qualitative analysis of the (potentially erroneous) outputs of contextualized embedding-based methods for detecting diachronic semantic change. First, we introduce an ensemble method outperforming previously described contextualized approaches. This method is used as a basis for an in-depth analysis of the degrees of semantic change predicted for English words across 5 decades. Our findings show that contextualized methods can often predict high change scores for words which are not undergoing any real diachronic semantic shift in the lexicographic sense of the term (or at least the status of these shifts is questionable). Such challenging cases are discussed in detail with examples, and their linguistic categorization is proposed. Our conclusion is that pre-trained contextualized language models are prone to confound changes in lexicographic senses and changes in contextual variance, which naturally stem from their distributional nature, but is different from the types of issues observed in methods based on static embeddings. Additionally, they often merge together syntactic and semantic aspects of lexical entities. We propose a range of possible future solutions to these issues.

1 Introduction

Lexical semantic change detection (LSCD) is a relatively recent sub-field within natural language processing. However, comprehensive surveys of data-driven modeling of diachronic semantic change are already available (Tang, 2018; Kutuzov et al., 2018; Tahmasebi et al., 2021a). Dedicated workshops on computational approaches to historical language change took place at the ACL conferences (Tahmasebi et al., 2019, 2021b, 2022) and the results of the SemEval-2020 Task 1 on unsupervised lexical semantic change detection were announced in March 2020 (Schlechtweg et al., 2020). Shared tasks for other languages followed soon (Basile et al., 2020; Kutuzov and Pivovarova, 2021).

The majority of the SemEval-2020 shared task participants employed methods based on word embeddings of various types. About half of them tried to make use of contextualized ('token-based') architectures like ELMo (Peters et al., 2018a) or BERT (Devlin et al., 2019). Although the winning systems still used non-contextualized ('static' or 'type-based') embeddings like word2vec (Mikolov et al., 2013), the difference in scores was not dramatic and we are most likely going to see more work in this direction. We agree with Schlechtweg et al. (2020) that as the contextualizing technologies mature, there will be a better understanding of how to properly use them for semantic change related tasks. Indeed, at the RuShiftEval shared task on LSCD for Russian (Kutuzov and Pivovarova, 2021), the leader-board was already dominated by contextualized models.

The current paper aims to contribute to this improved understanding by qualitatively analyzing the output of contextualized embedding-based approaches to the diachronic semantic change detection task. Hence, our work falls into the second category of ground truth semantic change evaluation, as defined by Hengchen et al. (2021): what is evaluated is the ranked output of the methods under investigation.

We here focus on Subtask 2 of SemEval-2020 Task 1: to rank a list of words by the degree of their semantic change between two historical corpora belonging to different time bins. The submissions were evaluated by their Spearman rank correlation against human annotations. This task was offered for four languages, each with their own word list and corpora: English, German, Latin and Swedish. One of the submissions in this Subtask was delivered by the UiO-UvA team (Kutuzov and Giulianelli, 2020). It used pre-trained ELMo models and achieved the average score of 0.37 at the evaluation phase (the second best contextual-

ized embedding-based system in this phase), and 0.62 at the post-evaluation phase (the best result overall in this phase). We chose their methods for closer inspection, because the implementations were publicly available, and the methods themselves are quite typical for the semantic change detection field (see below).

The contributions of this paper are twofold:

1. We propose a simple improvement to the approach in Kutuzov and Giulianelli (2020) by ensembling two of their best-performing methods. We show that it avoids the necessity to decide what method to choose, while still outperforming strong baselines.

2. We qualitatively examine the output of the contextualized methods for semantic change detection in English. We analyze examples of both correct and incorrect cases of detected semantic change. The latter findings are arguably more important for future studies, as one learns on errors. We propose a categorization of such problematic cases, relating them to inherent properties of pre-trained contextualized architectures in particular and distributional approaches in general.

2 Contextualized methods for detecting semantic change

Two methods for estimating semantic change were proposed in Kutuzov and Giulianelli (2020): PRT and APD (further detailed below). The methods are architecture-agnostic and can be used with any model able to produce contextualized token representations for a given sequence of word tokens. Overall, these methods can be considered typical representatives of using contextualized word embeddings for the task of semantic change detection: they boil down to directly comparing token embeddings of the target word in two periods; see (Martinc et al., 2020a) for a similar technique. Another possible approach (which we hope to analyze in the future) is clustering token embeddings into groups loosely corresponding to word senses and then comparing their time-specific distributions (Martinc et al., 2020b; Cuba Gyllensten et al., 2020; Giulianelli et al., 2020).

The common part of both the PRT and APD methods is as follows. Given two time periods t_1 and t_2, two corresponding corpora C_1 and C_2, and a set of target words, a language model (regardless of what it has been pre-trained on) is used to obtain contextualized token embeddings[1] of each occurrence of the target words in C_1 and C_2. Each target word w is then represented by two 'usage matrices' $\mathbf{U}_w^{t_1}$ and $\mathbf{U}_w^{t_2}$ consisting of all token embeddings produced for w. A *change score* is computed from these matrices, indicating the degree of semantic change undergone by a word between t_1 and t_2. The target words are ranked by this value. The methods differ in how exactly change scores are computed:

- **Inverted cosine similarity over word prototypes (PRT)**: the degree of change for w is calculated as the inverted cosine similarity between the average token embeddings ('prototypes') of all w occurrences in $\mathbf{U}_w^{t_1}$ and $\mathbf{U}_w^{t_2}$ correspondingly:

$$\mathrm{PRT}\left(\mathbf{U}_w^{t_1}, \mathbf{U}_w^{t_2}\right) = \frac{1}{c\left(\frac{\sum_{\mathbf{x}_i \in \mathbf{U}_w^{t_1}} \mathbf{x}_i}{N_w^{t_1}}, \frac{\sum_{\mathbf{x}_j \in \mathbf{U}_w^{t_2}} \mathbf{x}_j}{N_w^{t_2}}\right)} \quad (1)$$

  where $N_w^{t_1}$ and $N_w^{t_2}$ are the numbers of occurrences of w in time periods t_1 and t_2, and c is a similarity metric, for which we use cosine similarity. High PRT values indicate a higher degree of semantic change.

- **Average pairwise cosine distance between token embeddings (APD)**: the degree of change for w is measured as the average distance between all possible pairs of token embeddings in $\mathbf{U}_w^{t_1}$ and $\mathbf{U}_w^{t_2}$:

$$\mathrm{APD}\left(\mathbf{U}_w^{t_1}, \mathbf{U}_w^{t_2}\right) = \frac{1}{N_w^{t_1} \cdot N_w^{t_2}} \sum_{\mathbf{x}_i \in \mathbf{U}_w^{t_1},\ \mathbf{x}_j \in \mathbf{U}_w^{t_2}} d\left(\mathbf{x}_i, \mathbf{x}_j\right) \quad (2)$$

  where d is the cosine distance ($1 - c$ where c is cosine similarity). High APD values indicate a higher degree of semantic change.

Kutuzov and Giulianelli (2020) report that different test sets from the shared task manifested strong preference for either the PRT or the APD method, and that this is correlated with the distribution of gold scores in the test set (but not with its language). If the right method was chosen, then using contextualized embeddings to rank words by their degree of semantic change consistently outperformed the shared task baselines (frequency-based and count-based approaches) and the methods relying on type-based embeddings with orthogonal alignment (Hamilton et al., 2016a).

However, in a realistic setting it is obviously problematic to assume knowledge of the statistical properties of the target words beforehand. So, how should one choose between the PRT and APD methods? We found that simply averaging the PRT and APD estimates yields very robust predictions. In Table 1, we reproduce the results from Kutuzov and Giulianelli (2020), including the word2vec baseline, and add the 'PRT/APD' row

[1] Representations from the top layer of the model were used, since they yielded the best results according to Kutuzov and Giulianelli (2020).

Method	*English*	*German*	*Latin*	*Swedish*	*GEMS*	*Average*
SemEval-2020 Task 1 baselines						
FD (frequency difference)	-0.217	0.014	0.020	-0.150	0.068	0.094
CNT+CI+CD (count-based)	0.022	0.216	0.359*	-0.022	0.256*	0.166
Cosine distance with static embeddings (word2vec)						
Orthogonal Procrustes alignment	0.285	0.439*	0.387*	0.458*	0.235*	0.361
Contextualized embeddings						
BERT PRT	0.225	0.590*	**0.561***	0.185	**0.394***	0.391
BERT APD	0.546*	0.427*	0.372*	0.254	0.243*	0.368
BERT PRT/APD	0.498*	0.537*	0.431*	0.267	0.332*	0.413
ELMo PRT	0.254	**0.740***	0.360*	0.252	0.323*	0.386
ELMo APD	**0.605***	0.560*	-0.113	**0.569***	0.323*	0.389
ELMo PRT/APD	0.546*	0.678*	0.036	0.546*	0.360*	**0.433**
Inter-correlations between *ELMo* PRT and APD predictions						
Spearman's ρ	0.589*	0.655*	0.423*	0.538*	0.319*	0.505
Pearson's r	0.547*	0.656*	0.589*	0.698*	0.495*	0.597

Table 1: Spearman correlation with the gold standard per test set for the best methods from (Kutuzov and Giulianelli, 2020) and our PRT/APD ensemble approach. '*' denotes statistical significance of the correlation as measured by the two-sided p-value, $p < 0.05$.

with the scores we got using the ensemble approach. Note that in addition to the 4 shared task test sets, we also report results on the GEMS semantic change test set for English (Gulordava and Baroni, 2011). For individual test sets, the performance of PRT/APD usually lies in between PRT and APD, but when averaged over all five test sets, it ranks higher than any individual method, and this effect holds for both ELMo and BERT, with the best result yielded by ELMo. When compared to the shared task leader-board (Schlechtweg et al., 2020), the PRT/APD + ELMo combination outperforms all contextualized embedding-based systems in Subtask 2, supporting the same observation in (Kutuzov and Giulianelli, 2020).

Thus, the APD and PRT methods are complimentary, although their predictions are strongly correlated (see the bottom of Table 1). Together they act as a top-performing ensemble of the models, with the additional benefit of not having to worry about what method to choose. In the rest of this paper, we will use the PRT/APD method to produce semantic change scores for qualitative analysis. Note that since these scores are produced by an ensemble model, they are less interpretable than the original separate PRT and APD values. However, a manual inspection showed that the separate methods yield the same categories of errors as the combined score; see Section 5 below.

3 Data and models used

For our in-depth analysis of the results, we use textual data from the Corpus of Historical American English or COHA (Davies, 2012) (it is certainly desirable to reproduce this analysis for other languages, which we leave for future work). In particular, we deal with 5 COHA sub-corpora corresponding to five decades: the 1960s, 1970s, 1980s, 1990s and 2000s. Note that this setup is slightly different from the SemEval-2020 Task 1 in that we have a sequence of five time bins. With this, we aim to trace the lasting evolution of word meaning, not limited to changes between two time periods. The employed time span means we deal with relatively short-term meaning changes.

We chose ELMo as a contextualizer based on its better performance (Table 1) and much lower computational requirements than BERT. It allowed us to train a single model from scratch on the concatenation of all COHA texts belonging to the five decades mentioned above (the full corpus size is about 127 million word tokens, and we trained for 5 epochs). The texts were tokenized and lemmatized with the English UDPipe tagger trained on the Universal Dependencies 2.3 treebank (Straka and Straková, 2017), discarding punctuation marks and lower-casing all words.

The list of words to analyze is a concatenation of all words from the SemEval-2020 Task 1 English test set, all words from the GEMS test set, and 1000 randomly sampled words occurring in all five COHA sub-corpora with frequency in each sub-corpus higher than 100. Af-

ter excluding numerals, function words and the words with a total frequency of less than 1000 occurrences across all decades (to discard unstable representations of rare words), the resulting word list contains 690 entries. For each of them, we used our ELMo model to calculate their PRT/APD scores in the four consecutive pairs of the COHA decades (1960s–1970s, 1970s–1980s, 1980s–1990s, 1990s–2000s), thus producing a score matrix $\mathbf{M} \in \mathbb{R}^{690\times4}$. Below we examine the actual scores in this matrix, and how they are related to processes in the recent history of English.

4 Well-behaved examples

For many words, the scores do signal real changes, like a new emergent sense. Let us consider the word '*cell*' as an example. The dataset from Tsakalidis et al. (2019), based on the Oxford English Dictionary definitions, mentions it as having acquired a new sense of 'MOBILE PHONE' after 2000. Recall that PRT/APD produces as an output a measure of how strong the semantic change of a target word was between two time bins; this measure characterizes a pair of decades in our case. '*Cell*' received a change coefficient of 0.673 for the 1960–1970 pair (arguably corresponding to the start of its widespread usage in the biological sense).

After that, the estimated degrees of change were smaller, with 0.669 for 1970–1980s and 0.672 for 1980–1990s. However, 1990–2000s had a change coefficient of 0.695 (the highest for this word across all decades), most likely reflecting the new 'MOBILE PHONE' sense. As a side note, it might look like the PRT/APD values show very little variation: in fact the average standard deviation of $\mathbf{M}$ values across four time period pairs is 0.04, with the average PRT/APD value being about 0.70. This means that the change coefficients for '*cell*' are actually lower than the mean value in our dataset (z-score of 0.695 is −0.17). See more on this in the next Section 5.

Unlike the static word embedding approaches, using contextualized models allows one to visually explore the individual occurrences of a given word in different senses. For this purpose, we use Principal Component Analysis (PCA) to reduce the contextualized token embeddings of '*cell*' in our diachronic sub-corpora to their 2-dimensional projections. Figure 1 shows these projections for the decades from the 1970s through the 2000s.

Even at a glance, it is possible to see that in the 2000s, some radical changes in the groupings of the '*cell*' token embeddings occurred. The three previous decades are all characterized by a rather vague separation of this word's usages into two clusters (at the left and at the right part of the vector space). In the 2000s, we observe the appearance of a new cluster: now there are two strong clusters to the left and a third one to the right. But what senses do these clusters correspond to?

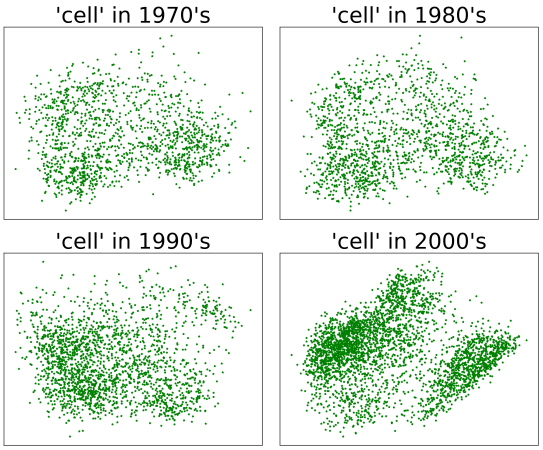


Figure 1: PCA projections of contextualized token embeddings of '*cell*' in four different decades.

Fortunately, since each point on the plot represents a particular '*cell*' occurrence from a particular decade's sub-corpus, we can retrieve their corpus contexts and manually inspect them. Of course, we did not inspect *all* occurrences: both due to their amount (thousands) and due to the absence of clear-cut cluster boundaries. Instead, we randomly sampled about 20 occurrences from the core area of each apparent cluster and examined them.[2]

We observe that in the 1970s, 1980s and 1990s, the right-hand cluster mostly contains sentences with '*cell*' in the sense of 'PRISON CELL', see example 1:

(1)
1. 'I'd known Archie Meltzer, the chief turnkey on duty, for over ten years, but you wouldn't have known it from the way he processed me for the **cells**.'
2. 'It also happened to me in a jail **cell**.'
3. 'If she had been writing to somebody in the darkness of her prison **cell**, what had she done with the message?'

The left cluster (stably increasing its relative size over time) mostly contains sentences with '*cell*' in the biological sense, with examples given in 2.

(2)
1. 'The sexual **cells** of Pyronema show this in ascomycetes.'
2. 'It's how a **cell** decides whether it becomes a muscle **cell** or a skin **cell**.'

[2]The same method is used below throughout the paper. In all cases when visible clusters appeared in the projections, they were strongly consistent, with at least 90% of the randomly sampled data points in a cluster belonging to a particular sense.

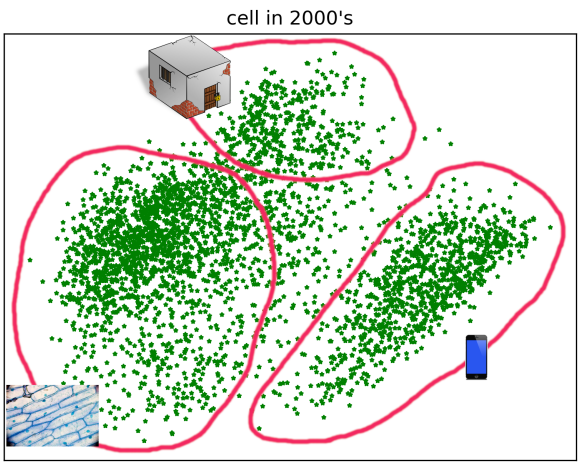


Figure 2: PCA projection of ELMo token representations of each occurrence of the word '*cell*' in the 2000s, with clusters labeled with senses.

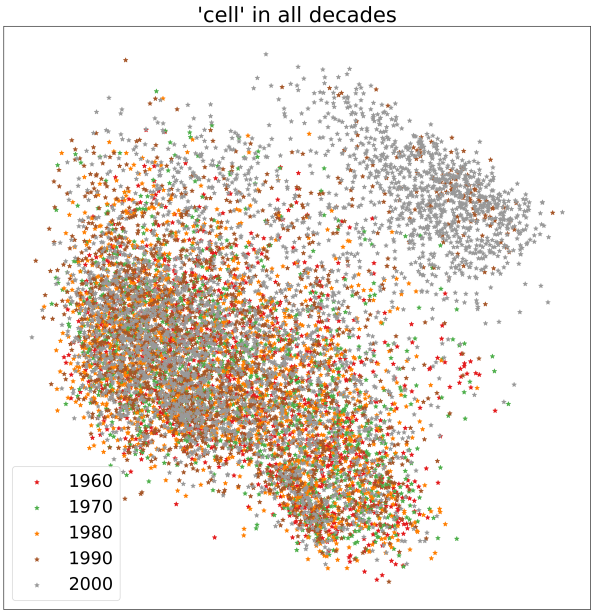


Figure 3: PCA projection of contextualized token embeddings of '*cell*' in all 5 COHA decades. Colors correspond to time periods.

3. 'If those **cells** are found to be cancerous after being sent to a lab, that's a definite diagnosis.'

After exploring the points in the 2000s plot in the same way, one observes that the two clusters on the left correspond to the old senses of '*cell*' (biological still at the bottom and prison at the top). But the new large cluster on the right almost exclusively consists of sentences mentioning '*cell*' in the sense of 'MOBILE PHONE' (see examples in 3 and Figure 2 displaying these clusters with labels).

(3)
1. 'But how well do the service providers fulfill that objective, and what about the other health and safety risks - exposure to radio waves and potentially fatal driver distraction - that the growing use of **cell** phones raise?'
2. '...he walked past, nearly dislodging the **cell** phone she had balanced between her chin and her left shoulder.'
3. 'You still have the same **cell** number.'

One can also visualize token embeddings for '*cell*' across all five time bins in one plot, as shown in Figure 3. Here, PCA dimensionality reduction is performed for all occurrences of this word (about 7500 total), and thus we can see how usages from different decades (shown in different colors) are grouped in relation to each other. The top right cluster is inhabited almost exclusively with the occurrences from the 2000s and to a less extent the 1990s. Not surprisingly, it contains sentences where '*cell*' is used in the 'MOBILE PHONE' sense. At the same time, in other parts of the plot, occurrences from all decades are distributed more or less uniformly, supporting our previous observation that in the 1960s, 1970s and 1980s, this word did not experience significant semantic changes.

In the case of '*cell*', the groupings of contextualized representations and the detected changes are undoubtedly connected to a new sense emerging (thus, a diachronic semantic shift). The relations between different senses of '*cell*' fall into the category of *homonymy*, where word senses are not directly related to each other (at least, synchronically). However, one can trace the cases of *polysemy* as well, where senses are synchronically related to each other. As an example, let us look at the adjective '*virtual*'. It experienced its strongest change of 0.769 in the 1980–1990 pair (its z-score is 1.9 in the full **M**).

Before 1990s, '*virtual*' was used mostly in two closely related senses: 'BEING SUCH IN ESSENCE OR EFFECT THOUGH NOT FORMALLY RECOGNIZED OR ADMITTED' (major one) and 'RELATED TO A HYPOTHETICAL PARTICLE WHOSE EXISTENCE IS INFERRED FROM INDIRECT EVIDENCE' (minor).[3] However, the 1990s saw the emergence of a large number of '*virtual*' usages in the sense of 'SIMULATED ON A COMPUTER OR COM-

[3]The definitions are taken from the Merriam-Webster dictionary (https://www.merriam-webster.com/).

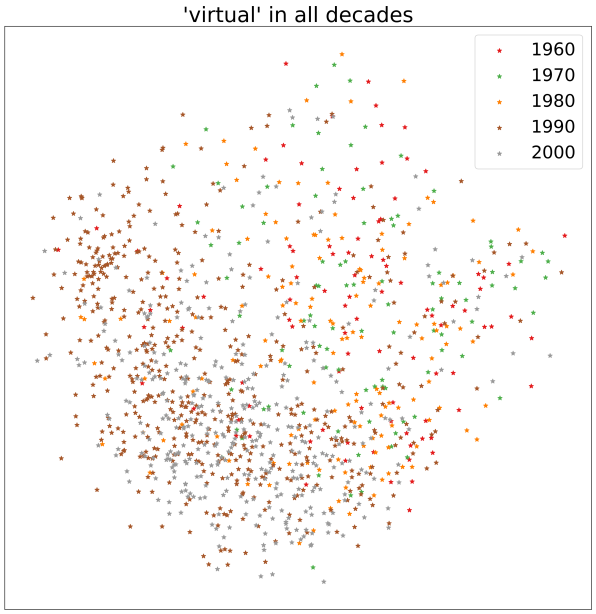


Figure 4: PCA projection of contextualized token embeddings of '*virtual*' in all 5 COHA decades.

puter network', especially in the expression 'virtual reality' (almost one third of all usages). This sense is related to the previous ones, thus manifesting a case of polysemy. The emergence of a new related sense in the 1990s is captured by contextualized embedding based methods, producing a higher change score for this time bin in comparison to the previous 1980s decade. We can also observe a much weaker change score of 0.740 in the 1990–2000 pair. The manual inspection of the occurrences shows that in the 2000s, '*virtual*' was still used a lot in this new third sense (interestingly, the 'virtual reality' expression itself almost came out of usage, constituting now only 6% of all '*virtual*' occurrences).

On the plot of '*virtual*' token embeddings across five COHA decades (Figure 4), the 'simulated on a computer or computer network' usages occupy the left part of the plot, with the 'virtual reality' phrases concentrated in the left top corner (as confirmed by manual inspection). The left part contains almost exclusively the occurrences from the 1990s and from the 2000s, while the left top corner is dominated by the 1990s.

So far so good: the contextualized embedding-based methods not only demonstrate high performance on the evaluation sets, they also produce interpretable predictions corresponding to well-known diachronic semantic shifts. But let us also look at a darker side of the **M** score matrix.

Word	Decade pair	Change	z-score
'*banish*'	1980s–1990s	0.794	2.60
'*designate*'	1980s–1990s	0.792	2.54
'*mg*' (m/gram)	1980s–1990s	0.791	2.52
'*progressive*'	1990s–2000s	0.782	2.27
'*indirectly*'	1990s–2000s	0.780	2.21
'*form*'	1990s–2000s	0.780	2.21
'*subsequently*'	1980s–1990s	0.780	2.21
'*neutral*'	1990s–2000s	0.779	2.18
'*traditionally*'	1990s–2000s	0.779	2.18
'*pointed*'	1960s–1970s	0.778	2.16

Table 2: 10 points with the highest change scores in 5 decades of COHA (as measured by PRT/APD). Z-scores are computed on the full **M**. Word color indicates its class, see Section 5.

5 Problematic examples

The picture is not as clear if one gets beyond hand-picked well-behaved examples. As mentioned above, the change coefficient of '*virtual*' when comparing the 1990s to the 1980s was 0.769. But the absolute values (and even z-scores) here are not very informative. There is no well-defined threshold: it is not the case that the change coefficients higher than, say, 0.7 always correspond to some breaking points in the word evolution. There are much stronger bursts which do not yield to such an explanation. Table 2 lists 10 words with the highest change coefficients in **M**. As can be seen, these changes are indeed unusually strong, all of them being more than 2 standard deviations away from the mean change score. However, none of them can be immediately interpreted as acquiring or losing a sense. What is the cause of these bursts?

5.1 Categories of problematic examples

Indeed, none of the 10 words with the highest scores is a schoolbook example of a semantic shift. We emphasize it does not necessarily imply outright errors or 'false positives'. As we show below, a good part of these words in fact do have reasons to be assigned high change scores; it is just that these reasons are somewhat different from what a historical linguist would expect to see.

Looking closely at these cases reveals three general word classes which trigger high semantic change score as measured by the PRT/APD approach, but at the same time did not undergo any semantic shifts in the classic understanding of the term (Bloomfield, 1933). The classes are (colors correspond to those in Table 2):

1. Words of strongly context-dependent meaning (‘*designate*’, ‘*progressive*’, etc): their token embeddings are very different from each other (and thus change scores are high) when compared either synchronically or diachronically.

2. Words frequently used in a very specific context in a particular time bin, different from other periods (‘*mg*’, ‘*indirectly*’, etc). It can be looked at either as a result of (unintended) domain shifting when building a corpus or as contextual variance which really exists in language, but did not yet lead to the emergence of a new lexicographic sense (or losing an old one). Note that Shoemark et al. (2019) observed very similar phenomena when analyzing Twitter data with static word embeddings. We will also call such cases ‘data bursts’. There is an interesting sub-type of this class:

    - words used as a proper name in a particular time bin (‘*banish*’, etc.); this leads to extremely high contextual variance and the emergence of isolated token clusters.

3. Words undergoing syntactic changes, not semantic ones; see below.

Note that the assignment of data points to classes in Table 2 was not done as a part of a full-fledged annotation effort with pre-defined error categories. Rather, this is a product of qualitative error analysis conducted by the authors: that is, the classes were identified as an attempt to group and systematize the problematic predictions of the methods used. We by any means do not claim that this grouping is the only one possible; however, as shown below, it models the data well enough to produce meaningful insights.

We remind the reader that the change coefficients were produced by the ensemble PRT/APD method. However, the PRT and APD methods on their own suffer from the same categories of problems. We analyzed 10 words with the highest estimated degree of change for the separate methods as well, and found them to largely overlap with those produced by PRT/APD; see Table 3. For APD, 60% of the points are the same words as for PRT/APD, for PRT it is 20%, but these two words are at the top of the list.[4]

An interesting observation is that each separate method tends to ‘favor’ different classes of problematic examples: while for PRT, seven words out of the top 10 are cases of data bursts (including the proper name subclass), for APD, nine of the top 10 are words with strongly context-dependent meaning. The PRT/APD method yields a more balanced distribution of these two classes (each takes approximately half of the top 10 list): this is arguably one of the reasons for its higher empirical performance. This aligns well with the assumption about the complementary nature of PRT and APD that we already mentioned before. The analysis of the reasons for this behavior is an interesting topic for future studies.

[4] Spearman ρ correlation between predictions of APD and PRT on $\mathbf{M}$ varies from 0.19 to 0.34, depending on a particular pair of decades; for Pearson, it is from 0.13 to 0.16; all the correlations are statistically significant.

PRT (score)	**Bin**	**APD** (score)	**Bin**
‘*mg*’ (1.17)	1990s	‘*designate*’ (0.57)	2000s
‘*banish*’ (1.11)	1990s	‘*progressive*’ (0.56)	2000s
‘*don*’ (1.11)	1980s	‘*form*’ (0.56)	2000s
‘*crunch*’ (1.07)	1970s	‘*subsequently*’ (0.55)	1970s
‘*immune*’ (1.07)	1980s	‘*lead*’ (0.55)	1990s
‘*clayton*’ (1.07)	1970s	‘*traditionally*’ (0.55)	2000s
‘*norm*’ (1.06)	1970s	‘*pointed*’ (0.55)	1970s
‘*brian*’ (1.06)	1970s	‘*truly*’ (0.55)	2000s
‘*ian*’ (1.06)	1980s	‘*mere*’ (0.55)	2000s
‘*sequence*’ (1.06)	2000s	‘*savage*’ (0.55)	2000s

Table 3: 10 points of the strongest change in 5 decades of COHA, as measured separately by PRT and APD. Word color indicates its class, see Section 5. ‘Bin’ columns denote the decade when the change occurred.

As a side note, two words predicted as changed by the PRT method do not fall into any of our categories: ‘*don*’ and ‘*immune*’. ‘*Don*’ stems from what seems to be a corpus pre-processing issue on the COHA side: in the 1980s sub-corpus of COHA, the frequency of ‘*don’t*’ tokenized as ‘*don ’ t*’ (with two spaces) is two orders of magnitude higher than in the other decades. This leads to the appearance of a very distinct ‘*don*’ cluster in this time bin. For ‘*immune*’, we observe that in the 1980s, it starts being actively used in the phrase ‘*immune system*’, again forming a separate cluster. This is not a temporary data burst, since it continued in the 1990s and in the 2000s. The dynamics of ‘*immune*’ is arguably related to the discovery of the HIV virus in the beginning of the 1980s, and thus, it can (cautiously) be acknowledged as a well-behaved example, not a problematic one. But let us return to the PRT/APD predictions.

Figure 5 shows the PCA projections of token embeddings for four of the words from Table 2 across the five COHA decades. Below we describe these diachronic vector spaces more closely to explain the nature of each category of ‘problematic’ words.

‘*Progressive*’ (in the bottom left part of the plot) belongs to the 1st class and presents the easiest case to explain. As can be seen from the plot, the occurrences from all five decades are spread uniformly over

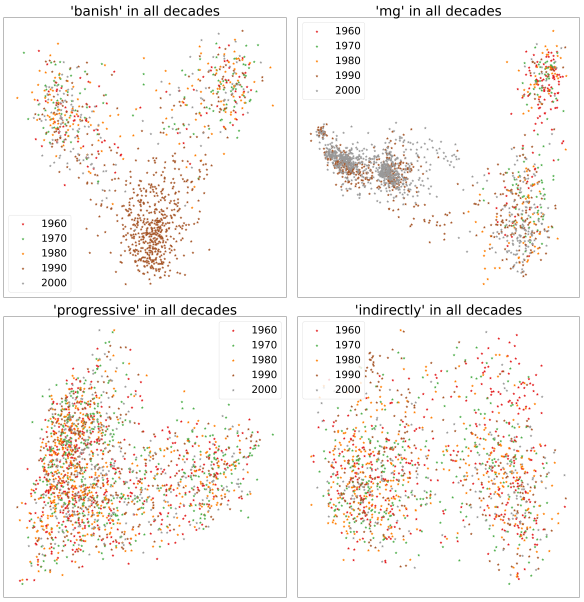


Figure 5: PCA projections of token embeddings for '*banish*', '*mg*', '*progressive*' and '*indirectly*' across all 5 COHA decades.

the vector space. There are no regions inhabited by occurrences only from some subset of the decades. This means no sense was acquired or lost at any point in time. The reason for the high absolute value of the change score is the context-dependent meaning of the word itself. Actually, it featured high change scores in all the previous decade pairs as well: 0.781, 0.780, 0.778. Its contexts are so diverse and 'fluid' that PRT/APD detects strong change whatever corpora are under comparison. In this respect, '*progressive*', '*designate*', '*form*' and similar entries behave much like function words: their contextualized embeddings are in a constant flux. Such cases can be traced and discarded when we have a sequence of several time bins clearly showing the constant character of the changes. However, if looking at one pair of time bins only (like in the SemEval 2020 Task 1), a researcher can be mistaken into concluding that an actual semantic shift is undergoing here.

'*Indirectly*' and '*mg*' (bottom and top right parts of the plot correspondingly) belong to the 2nd class and they do reflect some actual changes in the corpora. The plot for '*indirectly*' features a small cluster of the 1990s occurrences in the top left corner. Otherwise, the occurrences from different time bins are spread uniformly, so this must be the reason of the detected 'change'. Indeed, for this word we find high change coefficients both for the 1990s (0.779) and the 2000s (0.780), while before that the scores were much lower. Accordingly, something had happened to '*indirectly*' in the 1990s and then arguably went back to normal in the 2000s. Manual inspection of the 1990s-specific cluster reveals sentences like those in example 4:

(4)
1. 'Lane now holds 1,966,692 shares directly and **indirectly**, worth $ 17,700,228.'
2. 'Parshall now holds 300 Class A shares **indirectly**, worth $ 3,975.'

All of them are excerpts from a long text titled 'Depressed shares are a hit with bargain-hunting execs Banks, utilities among winners', apparently published in the 'Insider trading' magazine in 1994. It abounds with reports on various persons holding various amounts of shares directly or indirectly. This type of texts is unusual for COHA: there are no sentences mentioning both '*hold*' and '*indirectly*' simultaneously in other decades, except only one such sentence in the 1980s. Meanwhile, the 1990s sub-corpus has 27 of them (the size of the outlier cluster we see in the plot). The 2000s sub-corpus does not include such texts any more, and thus we observe an equally strong change back when moving from the 1990s to the 2000s.

For the word '*mg*' (milligram) the situation is similar, except that the change score of 0.792 in the 1990s was the only burst (for other decade pairs, the change scores do not exceed 0.71). It means that something changed in the 1990s and stayed like this through the 2000s. Inspecting Figure 5 (top right plot) shows that there is indeed a clearly separated cluster consisting only of the 1990s and 2000s tokens. In the corpus, they always occur in the phrase '*mg cholesterol*', in sentences like in example 5, being part of dish recipes.

(5) 'Per serving: 525 calories, 34 gm protein, ... 674 **mg cholesterol**, 6 gm saturated fat, 409 mg sodium',

'*Cholesterol*' did occur in COHA before the 1990s, but never in such a context (123 occurrences of '*mg cholesterol*' in the 2000s, 128 in the 1990s, and 0 before that).

In these cases, no semantic shifts in the mainstream sense of this term occurred: the word '*indirectly*' still had the same general meaning in the 1990s, and the word '*mg*' in the 1990s and 2000s. However, the PRT/APD method indeed detected anomalous contextual variances in the corpora under analysis. Another interesting case belonging to this type is the word '*neutral*', also appearing in Table 2. Its 2000s burst is caused by the emergence of the frequent collocation '*gender neutral*', which is missing (or extremely rare) in the previous decades. Are we observing a new sense gradually appearing, or is it just contextual fluctuation? Anyway, independent of whether these variances are

due to real changes in the word usage (caused by social and cultural developments) or due to improper corpus collection procedure, they are still really existing bursts in the data. In this respect, this type of controversial predicted changes is different from '*progressive*' or '*designate*'. This is another manifestation of a larger NLP problem of domain sensitivity (Okurowski, 1993). Essentially, what the model detected was a domain change in comparison to overall genre structure of COHA.

Finally, the word '*Banish*' belongs to the proper names subset of the 2nd class. It features a clearly separated cluster of token embeddings containing exclusively the 1990s occurrences (bottom of the plot). All of them are mentions of 'Banish' as the name of one of the characters of the 1996 novel 'The Standoff' by Chuck Hogan, see example 6:

(6)
1. '**Banish** slipped deeper into thought.'
2. '**Banish** smiled weakly at the sentiment.'
3. 'The sound man eyed him as he stepped inside, saying nothing about **Banish's** burnt face.'

The novel is included in COHA almost in its entirety, obviously bringing in a lot of '*banish*' usages very different from its mainstream verbal meaning (recall that we both lemmatize and lower-case our texts). This leads to the high change coefficients in the 1980–1990 pair: 0.794, a strong burst compared to 0.733 (1960s–1970s) and 0.730 (1970s–1980s). Note that the change score is high again when looking at the 1990–2000 pair (0.793). The obvious reason is that the 2000s corpus does not mention Banish from 'The Standoff' at all, so the meaning of '*banish*' has returned to its pre-1990s state (more or less equally distributed between the senses of 'to expel' and 'to destroy, to end').

Using '*Banish*' in this way is certainly creative, and even more importantly, these occurrences indeed denote something different from the regular meaning of '*banish*'. It can be disputed whether using a verb (or a common noun) as a proper name *is* coining a new sense. Note, however, that a very similar case of the word '*apple*' acquiring the new sense of a well-known company proper name is often used as a classic example for word sense disambiguation (Manion, 2014). From this point of view, '*banish*' certainly temporarily acquired a new sense in the COHA 1990s corpus, and thus the predicted change score perfectly reflects the reality. On the other hand, one could argue that this is true for the title-cased '*Banish*' only, but yielding high change score for '*banish*' is an error. See more on that in subsection 5.4.

During our manual analysis (following the same workflow of randomly sampling and examining about 20 usages from the core area of the cluster) we also observed multiple cases where token embedding clusters

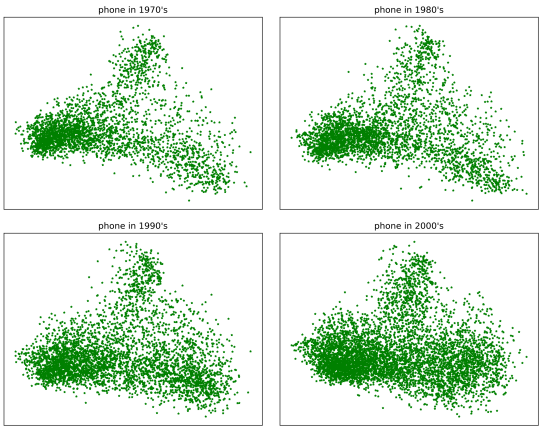


Figure 6: PCA projections of token embeddings for '*phone*' in four different decades: stable syntactic clusters.

of an unambiguous word manifested this word being used in different syntactic roles. For example, the word '*phone*' features three clusters of token embeddings, stable across time (Figure 6). They group occurrences not on *semantic*, but more on *syntactic* grounds:

1. '*phone*' is a subject: 'Then the **phone** rang.' (the top cluster)
2. '*phone*' is an object or an oblique argument: '...took a deep breath and grabbed the **phone**.' (the bottom left cluster)
3. '*phone*' is a modifier part of a compound noun: 'Please include a daytime **phone** number.' (the bottom right cluster)

This constitutes the 3rd class of problematic change predictions. If the syntactic role frequency distribution of a particular word changes diachronically, the change detection methods based on contextualized embeddings would be triggered by this. As a result, a syntactic shift will be taken for a semantic one. '*Traditionally*' from Table 2 is such an example: for some reason, the 1990s COHA sub-corpus contains much fewer usages of this word as an adjective modifier ('*traditionally christian*', '*traditionally male*', etc) than the other decades. Interestingly, this syntactic influence is expressed even though we extracted representations from the *top layer* of ELMo, which was shown by Peters et al. (2018b) to mostly contain *semantic* information. We discuss the possible smarter ways to employ the model layers in the subsection 5.4 below.

Word	Decade pair	Change	z-score
'*drew*'	1960s–1970s	0.892	4.19
'*banish*'	1980s–1990s	0.836	3.71
'*jessica*'	1960s–1970s	0.823	3.59
'*fanny*'	1960s–1970s	0.811	3.49
'*clayton*'	1970s–1980s	0.801	3.41
'*val*'	1970s–1980s	0.798	3.39
'*chris*'	1960s–1970s	0.790	3.32
'*max*'	1980s–1990s	0.760	3.07
'*joel*'	1980s–1990s	0.758	3.04
'*josh*'	1980s–1990s	0.743	2.92

Table 4: 10 points of the strongest change in 5 decades of COHA (as measured with static word embeddings).

5.2 What about static embeddings?

It can be argued that the issues mentioned above are not specific for contextualized architectures. To test this, we trained five static embedding models on five COHA sub-corpora each representing one of the decades (1960, 1970, 1980, 1990, 2000). We employed the widely used skip-gram with negative sampling (SGNS) algorithm from Mikolov et al. (2013), also known as *word2vec*. The training hyperparameters were set as follows: symmetric context window of 10 words to the right and 10 words to the left, minimal word frequency 5, vector size 300, 10 iterations over the corpus. Then we followed the standard semantic change detection workflow (so called 'SGNS+OP') :

1. Vector matrices of each model were aligned to the 2000s matrix with the Orthogonal Procrustes (OP) transformation (Hamilton et al., 2016b); the 2000s decade was chosen as the basis for alignment, since this model has the largest vocabulary (65 246 words).

2. For each target word, the cosine distances between its aligned static embeddings in the four consecutive pairs of the COHA decades were calculated. This resulted in the $\mathbf{M}_{static} \in \mathbb{R}^{690\times4}$ matrix, analogous to the $\mathbf{M}$ matrix for ELMo embeddings. The values in $\mathbf{M}_{static}$ are change scores inferred from the *word2vec* models.

Top ten change scores in $\mathbf{M}_{static}$ are shown in Table 4. Again, none of these words looks like an example of a genuine semantic shift, although their z-scores are even higher than those in Table 2. The important thing is that we observe only two words which also appeared at the top of $\mathbf{M}$: '*banish*' (PRT/APD and PRT) and '*clayton*' (PRT). Since static architectures do not yield token embeddings, one cannot analyze the underlying reasons for high change scores, as we did in the previous subsection. However, it is obvious that most (if not all) words at the top of $\mathbf{M}_{static}$ are proper nouns, which is fully in line with the findings in (Shoemark et al., 2019). This makes the predictions of the static models a bit more similar to those produced with the PRT method (which makes sense, since both PRT and static embeddings 'merge' all occurrences of a word into a single vector representation), but still substantially different from what any tested contextualized approach yields.

To some extent, the SGNS-OP predictions are potentially easier for 'de-noising': one simply has to filter out proper names, which is technically straightforward. Anyway, the take-away message here is that the majority of the problematic examples' categories we mentioned above indeed seem to be specific to contextualized architectures and not manifested in approaches based on static embeddings (which can have their own issues, of course).

5.3 Summarizing reflections

Although contextualized architectures are indeed promising for the tracing of diachronic semantic change (especially for finding supporting examples from the corpus), their usage is not entirely straightforward. When measuring the strength of lexical semantic change with contextualized embeddings, one should watch out for the three classes (and one sub-class) of possible unexpected results described above. A word occurrence can receive a very different token embedding not because the word has acquired a new sense, but because it is used in an unusual syntactic role, or because it is surrounded by unusual neighbors (for example, when the domain of the underlying texts has changed). Since the resulting semantic change score is a derivative of the arrays of token embeddings, one observes strong bursts which manifest changes in contextual variance of a word, not a semantic shift in the lexicographic meaning of this term. This is probably not what a historical linguist expects to see, although it can depend on the particular study and the working definition of 'semantic shift'.

Note that the problems described here are not entirely novel and have been discussed before in semantic change literature. They are also related to complicated questions about the nature of meaning and of what exactly it means to undergo a 'semantic shift', especially when we observe a case of contextual variance. If we stick to the distributional view that 'senses are in fact clusters of corpus usages' (Kilgarriff, 1997), the cases described above should definitely count as sense inventory changes, or at least the appearance of short-term senses which then fade away. If one does not employ external data sources (like ontologies or diachronic dic-

tionaries), there is no reliable way to discern 'semantic changes' from 'differences in the underlying textual data': they are simply the same thing.

All this is an inevitable consequence of accepting the data-driven distributional paradigm. It can be argued that any distributional corpus-based model suffers from these problems by definition, simply because it derives its signal from contexts surrounding word tokens. In fact, the 'clusters' on the plots in this section can be more properly described not as 'senses', but as 'sense nodules' ('lumps of meaning with greater stability under contextual changes') from Cruse (2000). However, it is now confirmed that this fundamental issue is still present in deep contextualized language models, often thought to be superior to their static type-based predecessors. Addressing it is a challenge facing the semantic change detection community in general. Before this issue is solved, the output of current semantic change detection models still needs human scrutiny, unless the downstream task at hand is tolerant to high amounts of false positives.

5.4 Possible remedies

This paper is aimed rather at results interpretation and analysis than at improving task scores. With this in mind, we here do not offer fully implemented and evaluated solutions addressing the issues described above. Still, in this subsection, some possible thought directions are outlined (they are by no means exhaustive).

The 1st class (words with 'fluid' meaning) is clearly erroneous. These words always exhibit strong change without it being of any significant linguistic interest, and ways must be devised to filter out these cases. Possible approaches to do this could include measuring change scores between random subsets of the same time bin: if they are as high as those between different time bins, the possible reason is the word's fluidity, not real semantic change.

The 2nd class ('data bursts') can be considered erroneous or not, depending on one's definition of semantic change (e.g., whether it includes contextual variance). It can be looked at as a corpus problem: COHA is not entirely well-balanced with respect to sense distribution. On the other hand, any dataset is biased and incomplete, and the notion of a '100% balanced' corpus is in fact ill-defined (balanced *for what*?). Arguably, the creators of COHA did not set an aim to somehow 'properly represent' the distribution of word senses (even if there existed robust methods to implement this). As Hengchen et al. (2021) put it, 'whatever is encountered in corpora is only valid for those corpora and not for language in general'. For the subclass of proper names, pre-processing decisions can help: keeping proper names capitalized will avoid them mixing with common nouns and predicting a shift for an otherwise stable noun which just happens to have a popular proper name counterpart. On the other hand, this raises difficult questions about the boundaries between word types and about the correctness of separating 'Apple' from 'apple' based on their written forms only. Again, what constitutes an error here has to be decided separately for each particular study.

To detect the cases belonging to the 3rd class (syntactic shifts), one can arguably use the distributions of PoS tags surrounding a given word. However, this approach is not scalable except for the cases when we are interested in a small closed set of target words only. Another option is learning a weighted function of different layers of the language model (both lower layers carrying more syntactic information and higher layers carrying more semantic information) to properly discern between changes on different language tiers.

In any case, this will require a human annotated dataset of changes of different types. With this at hand, it will be possible to train a meta classifier taking as an input the PRT and APD change coefficients (including signals from different network layers), frequency values, capitalization and other features mentioned above and producing a binary decision on whether the current data point is potentially a false positive.

6 Limitations

Our analysis in Section 5 was based on the top 10 most changed words according to each change detection method. We acknowledge that more insights can be obtained by analyzing more top ranking words (this is also true for static embeddings).

Another important limitation of this work is our focus on false positives: that is, words which are assigned a high semantic change score when this arguably should not be the case. The study of false negatives (words known to have changed but assigned low scores by the models) is a topic of its own. It is related to possible analysis of the PRT, APD and PRT/APD predictions on the 'stable' versus 'changed' words from the SemEval-2020 test set (Schlechtweg et al., 2020). We hope to deal with these aspects in the future.

The plots in Sections 4 and 5 show token representations of our target word. A potentially more powerful visualization approach could include showing also some 'anchor' or 'seed' words serving to better disambiguate senses of different tokens (or time-dependent representations for static word embeddings). Note, however, that choosing such anchor words is a separate task in itself, see, for example, Hamilton et al. (2016b). In addition, the plots could arguably be made more visually enticing and insightful by using different markers and sub-sampling of data points (to make the plots look cleaner). This was out of scope for this work.

7 Conclusion

We have qualitatively analyzed the outputs of contextualized embedding-based methods for detecting diachronic semantic change. First, we improved the results of prior work by proposing an ensemble of two methods from Kutuzov and Giulianelli (2020), which proved to be a robust solution across the board, outperforming prior contextualized methods on the SemEval-2020 Task 1 test sets (Schlechtweg et al., 2020) and on the GEMS test set (Gulordava and Baroni, 2011). Our 'PRT/APD' method is more suitable for a realistic case of not knowing the gold score distribution beforehand.

Using PRT/APD together with ELMo, we produced semantic change coefficients for 690 English words across five decades of the 20 and 21 century using the COHA corpus (Davies, 2012), and systematically examined these predictions. Although many cases of strong detected change do correspond to well-known semantic shifts, we also found multiple less clear-cut cases. These are the words for which a high change score is produced by the model, but it is not related to any 'proper' diachronic semantic shift (not causing a new entry in a dictionary). We discuss such cases in detail with examples, and propose their linguistic categorization. Note that these issues do not depend on a particular training algorithm (or an ensemble of algorithms). There is no reason for them to not appear also when using BERT or any other token-based embedding architecture; see Giulianelli et al. (2020) and Yenicelik et al. (2020) who show that BERT generates representations which form structures tightly coupled with syntax and even sentiment. To properly test it empirically could be an interesting future work, but we have already shown that semantic change detection approaches based on static word embeddings (as opposed to contextualized token-based architectures) yield different sorts of problematic predictions.

It is not immediately clear whether improving the quality and representativeness of diachronic corpora can help alleviating this issue (producing more historical data is often not feasible if not impossible). Still, it would be interesting to refine our results using larger or cleaner historical corpora: for example, Clean COHA (Alatrash et al., 2020). We also plan to analyze the semantic change modeling results for other languages besides English, as well as using different neural network layers to infer semantic change predictions.

The data (change scores for all target words) and code (including visualization tools) used in this work is available at `https://github.com/ltgoslo/lscd_lessons`.

References

Alatrash, Reem, Dominik Schlechtweg, Jonas Kuhn, and Sabine Schulte im Walde. 2020. CCOHA: Clean corpus of historical American English. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 6958–6966, Marseille, France. European Language Resources Association.

Basile, Pierpaolo, Annalina Caputo, Tommaso Caselli, Pierluigi Cassotti, and Rossella Varvara. 2020. DIACR-Ita @ EVALITA2020: Overview of the EVALITA2020 Diachronic Lexical Semantics (DIACR-Ita) Task. *Proceedings of the Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2020) CEUR Workshop Proceedings (CEUR-WS.org)*.

Bloomfield, Leonard. 1933. *Language*. Allen & Unwin.

Cruse, D Alan. 2000. Aspects of the micro-structure of word meanings. In *Polysemy: Theoretical and computational approaches*, pages 30–51. OUP.

Cuba Gyllensten, Amaru, Evangelia Gogoulou, Ariel Ekgren, and Magnus Sahlgren. 2020. SenseCluster at SemEval-2020 task 1: Unsupervised lexical semantic change detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 112–118, Barcelona (online). International Committee for Computational Linguistics.

Davies, Mark. 2012. Expanding horizons in historical linguistics with the 400-million word corpus of historical american english. *Corpora*, 7(2):121–157.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Giulianelli, Mario, Marco Del Tredici, and Raquel Fernández. 2020. Analysing lexical semantic change with contextualised word representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3960–3973, Online. Association for Computational Linguistics.

Gulordava, Kristina and Marco Baroni. 2011. A distributional similarity approach to the detection of semantic change in the Google books ngram corpus. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 67–71,

Edinburgh, UK. Association for Computational Linguistics.

Hamilton, William, Jure Leskovec, and Dan Jurafsky. 2016a. Cultural shift or linguistic drift? Comparing two computational measures of semantic change. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2116–2121. Association for Computational Linguistics.

Hamilton, William, Jure Leskovec, and Dan Jurafsky. 2016b. Diachronic word embeddings reveal statistical laws of semantic change. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1489–1501, Berlin, Germany. Association for Computational Linguistics.

Hengchen, Simon, Nina Tahmasebi, Dominik Schlechtweg, and Haim Dubossarsky. 2021. *Computational approaches to semantic change*, chapter Challenges for computational lexical semantic change. Language Science Press.

Kilgarriff, Adam. 1997. I don't believe in word senses. *Computers and the Humanities*, 31(2):91–113.

Kutuzov, Andrey and Mario Giulianelli. 2020. UiO-UvA at SemEval-2020 task 1: Contextualised embeddings for lexical semantic change detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 126–134, Barcelona (online). International Committee for Computational Linguistics.

Kutuzov, Andrey, Lilja Øvrelid, Terrence Szymanski, and Erik Velldal. 2018. Diachronic word embeddings and semantic shifts: a survey. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1384–1397, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Kutuzov, Andrey and Lidia Pivovarova. 2021. RuShiftEval: a shared task on semantic shift detection for Russian. *Computational linguistics and intellectual technologies: Papers from the annual conference Dialogue*.

Manion, Steve Lawrence. 2014. *Unsupervised Knowledge-based Word Sense Disambiguation: Exploration & Evaluation of Semantic Subgraphs*. Ph.D. thesis, University of Canterbury. Department of Mathematics & Statistics.

Martinc, Matej, Petra Kralj Novak, and Senja Pollak. 2020a. Leveraging contextual embeddings for detecting diachronic semantic shift. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4811–4819, Marseille, France. European Language Resources Association.

Martinc, Matej, Syrielle Montariol, Elaine Zosa, and Lidia Pivovarova. 2020b. Capturing evolution in word usage: Just add more clusters? In *Companion Proceedings of the Web Conference 2020*, pages 343–349.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems 26*, pages 3111–3119.

Okurowski, Mary Ellen. 1993. Domain and language evaluation results. In *Fifth Message Understanding Conference (MUC-5): Proceedings of a Conference Held in Baltimore, Maryland, August 25-27, 1993*.

Peters, Matthew, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018a. Deep contextualized word representations. 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 2227–2237. Association for Computational Linguistics.

Peters, Matthew, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018b. Dissecting contextual word embeddings: Architecture and representation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509, Brussels, Belgium. Association for Computational Linguistics.

Schlechtweg, Dominik, Barbara McGillivray, Simon Hengchen, Haim Dubossarsky, and Nina Tahmasebi. 2020. SemEval-2020 task 1: Unsupervised lexical semantic change detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1–23, Barcelona (online). International Committee for Computational Linguistics.

Shoemark, Philippa, Farhana Ferdousi Liza, Dong Nguyen, Scott Hale, and Barbara McGillivray. 2019. Room to Glo: A systematic comparison of semantic change detection approaches with word embeddings. 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 66–76, Hong Kong, China. Association for Computational Linguistics.

Straka, Milan 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. Association for Computational Linguistics.

Tahmasebi, Nina, Lars Borin, and Adam Jatowt. 2021a. *Survey of computational approaches to lexical semantic change detection*. Language Science Press.

Tahmasebi, Nina, Lars Borin, Adam Jatowt, and Yang Xu. 2019. *Proceedings of the 1st International Workshop on Computational Approaches to Historical Language Change*. Association for Computational Linguistics, Florence, Italy.

Tahmasebi, Nina, Adam Jatowt, Yang Xu, Simon Hengchen, Syrielle Montariol, and Haim Dubossarsky, editors. 2021b. *Proceedings of the 2nd International Workshop on Computational Approaches to Historical Language Change 2021*. Association for Computational Linguistics, Online.

Tahmasebi, Nina, Syrielle Montariol, Andrey Kutuzov, Simon Hengchen, Haim Dubossarsky, and Lars Borin, editors. 2022. *Proceedings of the 3rd Workshop on Computational Approaches to Historical Language Change*. Association for Computational Linguistics, Dublin, Ireland.

Tang, Xuri. 2018. A state-of-the-art of semantic change computation. *Natural Language Engineering*, 24(5):649–676.

Tsakalidis, Adam, Marya Bazzi, Mihai Cucuringu, Pierpaolo Basile, and Barbara McGillivray. 2019. Mining the UK web archive for semantic change detection. In *Proceedings of Recent Advances in Natural Language Processing conference*.

Yenicelik, David, Florian Schmidt, and Yannic Kilcher. 2020. How does BERT capture semantics? a closer look at polysemous words. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 156–162, Online. Association for Computational Linguistics.