UNIVERSITY OF CALIFORNIA SAN DIEGO

A Linguistic Approach to Crosslingual and Multilingual NLP

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Linguistics with a Specialization in Computational Social Science

by

Catherine Arnett

Committee in charge:

Professor Farrell Ackerman, Chair
Professor Benjamin K. Bergen
Professor Leon Bergen
Professor Victor Ferreira

2025

The dissertation of Catherine Arnett is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2025

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank my committee. Farrell took me on as his student in the middle of the PhD and has since provided a great deal of intellectual and emotional support, without which I would not have finished this dissertation. Our discussions allowed me to explore a variety of paths and consider inspirations from a variety of disciplines, which has enriched my work greatly. Thank you to Ben for providing an ideal environment for me to foster new research interests and learn about a totally new field, both through lab meetings and our individual meetings. I want to thank Leon, whose classes were a critical component of my training and shaped my academic and career trajectories. I also thank Vic for hosting me in his lab for several years. Lab meetings were valuable for continuing to train me in rigorous experimental design and theoretically impactful work. I thank all members of the committee for being flexible and willing to support this dissertation after its topic changed drastically.

I want to thank the members of the Language and Cognition Lab, who have provided support and inspiration for my work, as well as invaluable collaboration. In particular, I'd like to thank Tyler for teaching me nearly everything I know and for being an enabler of ambitious projects.

I'd like to thank all the members of the Linguistic department, in particular the other graduate students, whose sage advice I sometimes followed. I want to thank Ebru, who a companion through huge changes and uncertainty. Thank you to all the regular attendees at department happy hours. And thanks to my cohort, who have been so supportive over the last nearly six years.

Thanks to members of the Language Production Lab for thoughtful discus-

sions.

Thanks to my family, who have been so understanding and patient.

Thanks to all my friends, especially MK, Louisa, Lisa, Jess, Dotty, Ben, Mia, Stephan, Akshay, Will, Phil(ip), Nojan, Olivia, Andy.

Finally I want to thank the hot tubs and Mesa Nueva and everyone who has ever hung out there with me. Conversations there are largely responsible for me not giving up.

Thank you to my husband, James, who has supported me at every step. I would not be where I am now without his patience and encouragement.

Thank you to all my co-authors, without whom none of these papers would have ever been completed.

Chapter 2, in full, is a reprint of the material as it appears in Catherine Arnett, Tyler A. Chang, Benjamin K. Bergen (2024). A Bit of a Problem: Measurement Disparities in Dataset Sizes Across Languages. 3rd Annual Meeting of the Special Interest Group on Under-resourced Languages (SIGUL), co- located at LREC-COLING. Torino, Italy.

Chapter 3, in full, is a reprint of the material as it appears in Catherine Arnett, Pamela D. Rivière, Tyler A. Chang, and Sean Trott (2024). Different Tokenization Schemes Lead to Comparable Performance in Spanish Number Agreement. Special Interest Group on Computational Morphology and Phonology (SIGMORPHON) co-located at NAACL. Mexico City, Mexico.

Chapter 4, in full, is a reprint of the material as it appears in Catherine Arnett and Benjamin K. Bergen (2025). Why do language models perform worse for mor-

phologically complex languages? Proceedings of the 31st International Conference on Computational Linguistics (COLING).

Chapter 5, in full, is a reprint of the material as it appears in James A. Michaelov, Catherine Arnett, Tyler A. Chang, Benjamin K. Bergen (2023). Structural priming demonstrates abstract grammatical representations in multilingual language models. Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP). Singapore.

VITA

2018	MA in Chinese and Linguistics (Hons), University of Edinburgh
2025	Ph. D. in Linguistics with a Specialization in Computational Social Science, University of California San Diego

PUBLICATIONS

Catherine Arnett and Benjamin K. Bergen (2025). Why do language models perform worse for morphologically complex languages? Proceedings of the 31st International Conference on Computational Linguistics (COLING).

Pavel Chizhov*, **Catherine Arnett***, Elizaveta Korotkova, Ivan P. Yamshchikov (2024). BPE Gets Picky: Efficient Vocabulary Refinement During Tokenizer Training. Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP). Miami, FL, USA. *equal contribution.

Tyler A. Chang, **Catherine Arnett**, Zhuowen Tu, Benjamin K. Bergen (2024). When Is Multilinguality a Curse? Language Modeling for 250 High- and Low-Resource Languages. Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP). Miami, FL, USA.

Jesse Quinn, Matthew Goldrick, **Catherine Arnett**, Victor S. Ferreira, Tamar H. Gollan (2024). Syntax Drives Default Language Selection in Bilingual Connected Speech Production. Journal of Experimental Psychology: Learning, Memory, and Cognition.

James Michaelov, **Catherine Arnett**, Benjamin K. Bergen (2024). Revenge of the Fallen? Recurrent Models Match Transformers at Predicting Human Language Comprehension Metrics. The First Conference on Language Modeling (COLM). Philadelphia, USA.

Catherine Arnett*, Pamela D. Rivière*, Tyler A. Chang, and Sean Trott (2024). Different Tokenization Schemes Lead to Comparable Performance in Spanish Number Agreement. Special Interest Group on Computational Morphology and Phonology (SIGMORPHON) co-located at NAACL. Mexico City, Mexico. *equal contribution

Catherine Arnett*, Tyler A. Chang*, Benjamin K. Bergen (2024). A Bit of a Problem: Measurement Disparities in Dataset Sizes Across Languages. 3rd Annual Meeting of the Special Interest Group on Under-resourced Languages (SIGUL), co-located at LREC-COLING. Torino, Italy. *equal contribution

James A. Michaelov*, **Catherine Arnett***, Tyler A. Chang, Benjamin K. Bergen (2023). Structural priming demonstrates abstract grammatical representations in multilingual language models. Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP). Singapore. *equal contribution

Catherine Arnett (2019). Pathways of Change in Romance Motion Events: A Corpus-based Comparison. *Proceedings of the Thirtieth Western Conference on Linguistics (WeCOL)*. Vol 23. Fresno, CA, USA.

ABSTRACT OF THE DISSERTATION

A Linguistic Approach to Crosslingual and Multilingual NLP

by

Catherine Arnett

Doctor of Philosophy in Linguistics with a Specialization in Computational Social Science

University of California San Diego, 2025

Professor Farrell Ackerman, Chair

Language models work well in English, but in just about every other language, they work much worse. In this dissertation, I use theories and methods from linguistics and psycholinguistics to contribute to the understanding of how language models work for different languages and how they work in multilingual settings. As languages differ greatly in how they encode information, many researchers have asked to what extent those crosslinguistic differences impact language model performance. I investigate the role of training data size and tokenizers in those differences. I find

that crosslinguistic differences which have been described in terms of typological features can instead be attributed to differences in effective dataset size.

In multilingual settings, language models may use some of the same representations to encode information for multiple languages. This allows for efficient usage of the models' parameters, while also yielding benefits for the models' ability to generalize across and between languages. I use a psycholinguistic experimental paradigm, crosslinguistic structural priming, to probe these shared representations and characterize how and when models learn these representations. These results also contribute to our understanding of how bilingual people use shared representations to store information about multiple languages.

Chapter 1

Introduction

This dissertation identifies key issues in natural language processing (NLP) as applied to a wide variety of languages. The field has long been English-centric, therefore, it is not well understood how language processing techniques developed for English generalize to other languages or how they work in multilingual settings. In this dissertation, I investigate two factors which contribute to crosslingual language model development: data and tokenizers. I use an experimental paradigm from psycholinguistics to characterize crosslingual transfer, which is the underlying mechanism by which language models efficiently represent multiple languages.

In the introductory chapter, I provide an overview of language models, introducing the necessary background information without assuming prior knowledge of language models. I define two distinct areas of research: crosslingual and multilingual NLP. I introduce key questions and issues within those areas. These questions relate to the broader goal of improving language equity in NLP, such that language

technologies can work better for more languages.

This is a staple thesis, where most chapters have been published independently of one another. This introductory chapter and other accompanying text provides context and necessary background for each chapter. These introductions highlight the common themes of the chapters and discuss the impact of the works, as well as developments that have occurred since the papers were originally published.

1.1 Introduction to Language Models

A language model is a probabilistic model of natural language (Jurafsky and Martin, 2024). While ultra-large Transformer models like the GPT models, e.g GPT-3, are perhaps the most well-known examples of language models know, the term language model encompasses n-grams, RNNs and LSTMs, as well as models like GPT-3. At their core, all of these models take as input a large body of text. They transform the text into vast matrices of numbers. The resulting model is able to process strings of natural language and generate text completions according to the distributional statistics of the training data.

It is helpful to think of language models as a word calculator (Willison, 2023). These models take a string of natural language, convert it into a series of token ids, which each have a representation stored in the model (token embeddings). Then, these models do a large number of matrix multiplication operations over the token representations. This yields the probability distribution over the model's entire vocabulary. The model predicts the next token by sampling from that distribution

utilizing various methods, such as selecting the item with the highest assigned probability.

The core principles of language models can be introduced with n-gram language models, which have existed for nearly 80 years (Shannon, 1948). The mathematical foundations have existed for much longer. Modern Large Language Models fundamentally operate on the same principles, but are able to better handle much longer strings and better predict novel strings. An *n-gram* refers to a sequence of length n. An n-gram language model predicts a token based on its co-occurrence frequency with other words. A 2-gram (more frequently called bigram) model uses sequences of length two, a 3-gram (trigram) models use sequences of three, a 4-gram uses sequences of four, and so on. There are also unigrams (sequence length of 1), but if the sequence length is 1, then in practice it is a word frequency list. A bigram model will predict a word based on the word before it. A trigram model will predict a word based on the two previous words, et cetera.

N-gram frequencies are calculated using a sliding window of length n. The number of occurrences of each sequence of length n For example, if n=2, and the following sentence acts as an example corpus,

```
The woman sat on the bench by the pond on the end of the path.
```

The bigram frequencies are:

```
(The woman) - 1
(woman sat) - 1
(sat on) - 1
(on the) - 2
...
```

The first two words in the corpus are (The woman). This bigram is assigned count of 1. The next two-word sequence is (woman sat). This bigram is also assigned a count of 1. As the window of length slides along the text, it reaches the bigram (on the). It is initially assigned a count of 1. As the window keeps sliding, the same bigram appears again. The count for this bigram is now 2.

To use these counts to predict the next word, we can calculate the conditional probability of each following token given the previous token in the n-gram window. In this example, the model is a bigram model, so it predicts the next token based only on the previous word.

For the following context, the bigram model will predict the next token based on the word 'the'.

```
The book on the ___
```

There are three bigrams in our example model that begin with 'the':

```
(the bench)
(the pond)
(the path)
```

We can normalize the frequency of each of these bigrams by the total frequency of all bigrams that begin with 'the', which is three. Each of these bigrams has a count of 1, so each of their relative frequency is $\frac{1}{3}$. The relative frequency is also equivalent to the conditional probability of the second token in each bigram given the previous token, 'the'. This can be expressed:

```
P(bench|the) = 0.33
P(pond|the) = 0.33
P(path|the) = 0.33
```

The model can now sample from this probability distribution. Note that there are only three possible completions, because these are the only word following 'the' that the model has seen. The model assigns a probability of 0 to every other word that could follow 'the'. A greedy sampling method would mean the model chooses the next token with the highest relative probability. For this example, the probability of each continuation is the same. In practice, as training corpora are much larger than one sample, it is less common that all of the possible completions will be exactly the same. However, when this situation arises, a model may fall back on another metric to predict the next token. One of these methods is called *backoff*. In this method, the model uses lower-order n-grams to predict the next word. In a bigram model, the only lower-order n-gram is a unigram language model. For a bigram model, therefore, backoff may entail using the oveall word probability to predict the next word. In this case, the model would then select whichever of the three candidates has a higher word frequency in the training corpus. In a higher-order n-gram, like a 5-gram, backoff might entail using 4-gram, then trigram, then bigram frequencies. This is one method language models might use to predict tokens in various contexts.

Models predict the next word in a sequence based on the context. The longer the context, generally the easier it is to predict the next word. Therefore, increasing n will often improve the language model's performance, as the context is richer and the set of possible answers is more constrained. The basic intuition is reflected in human language processing. The cloze task (Taylor, 1953), which is extremely similar to the next-word-prediction task that n-grams are doing, is a task in which human participants provide sentence completions given a particular context. This experi-

mental paradigm has been used to study various aspects of language comprehension and production, as it is correlated with metrics of human language processing like N400 (Kutas and Hillyard, 1984) and reading time (Smith and Levy, 2011). It has been shown that more constrained contexts lead to higher-cloze responses. That is, the more constrained the context, the more participants converge on the same completion.

For example, in the two sentences below, the predictions for completions for (1-a) are more constrained than those for (1-b). Many people would probably guess the completion for (1-a) is 'muscle', but there are many plausible completions for (1-b), e.g. 'person studying language models is just doing it because of hype' or 'trip to London isn't complete without seeing London Bridge'.

(1) a. The athlete pulled a ___

b. Peter says that a ___

(Staub et al., 2015)

For the same reasons, higher-order n-gram models are likely to make better predictions than lower-order models. As n increases, however, n-gram models are limited by data sparsity. An n-gram model will assign a probability of 0 to an unseen n-gram. The longer the n-grams, the lower each of their frequencies will be and the more n-grams the model could encounter would be assigned a probability of 0. Therefore, at a certain point, a larger n may hamper an n-gram model's ability to predict the next word. Contemporary language models are much better at handling this limitation than n-gram models, as they better learn patterns that can generalize to new

contexts.

1.1.1 Tokenization

Language models cannot process raw strings of natural language data. They operate over **tokens**, which are discrete chunks that are contained in the language model's vocabulary. The vocabulary is determined by the tokenizer, which contains a list of tokens which can be used to represent tokenized text.

Language models use tokenizers because otherwise they would have to operate over words. Aside from being a fraught term with no accepted linguistic definition[1], there are several practical reasons why this is not possible. For a language like English, one operationalization of wordhood is as sequences separated by whitespaces. However, English has hundreds of thousands of unique word forms. Dictionaries such as Merriam-Webster and The Oxford English Dictionary each have approximately half a million unique entries. This does not include inflectional variants, e.g. *write*, *writes*, *wrote*. Nor does this include variation introduced by capitalization, misspelling, or non-standard orthography. For a language like English, it could require millions of vocabulary entries to capture every unique whitespace-separated string in a language model training corpus. Whitespace-separated words are even more problematic for languages like Chinese, Japanese, and Korean. These languages do not put spaces between orthographic words. There may not be any spaces in a paragraph of text. Therefore, defining units of text with whitespace separation is problematic

[1]There is no widely accepted definition of 'word' as a term in Linguistics (Haspelmath, 2023). See Hall (1999) for an overview of phonological wordhood and Haspelmath (2017) for a discussion of criteria for defining the morphosyntactic word.

for multiple reasons.

Tokenizers are trained on a corpus of natural language data – usually a very small subset of the data that the language model is trained on. Different tokenizers are trained differently, so I will take Byte-Pair Encoding (BPE) tokenization as an example, as it is the most widely used type of tokenizer. The tokenizer functions by breaking down text into the smallest units and then doing binary merges until the result is a list of items in the tokenizer vocabulary.

To train the tokenizer, first, the text is broken down into each byte that composes the text. For illustration purposes, we can think of the text as being broken down into characters. So, if the tokenizer was being trained on the text, "The cat in the hat.", the first step of tokenizer training would be to break down the text:

```
T h e  c a t  i n  t h e  h a t .
```

After this step, each unique character is stored in the tokenizer vocabulary.

Next, the tokenizer identifies the most frequent bigram (sequence of two characters). In this case, it is `(h, e)`. This is the first merge stored by the tokenizer. The resulting text is:

```
T he  c a t  i n  t he  h a t .
```

After each merge, the resulting string is stored as a vocabulary item, as merges can only occur between two items in the tokenizer vocabulary.

The next merge is `(a, t)`, which results in:

```
T he  c at  i n  t he  h at .
```

The tokenizer continues training until it either runs out of slots to fill in its vocabulary, which is a pre-specified number, or there are no more possible merges. The tokenizer consists of a list of vocabulary items (tokens), and the list of merges. The tokenizer stores these merges in the order in which they were learned. Each of these tokens is assigned a **token id**, which is determined by the order in which the token was added to the tokenizer vocabulary. The vocabulary items with lower token ids were learned earlier in tokenizer training. The number of token ids is equivalent to the vocabulary size of the tokenizer.

During tokenizer inference, when the tokenizer is used to tokenize a text, the tokenizer splits the text into characters and then executes the merges in the order that they were learned. This makes the tokenizer extremely computationally efficient and fast. In addition to converting strings of natural text into a form the language model can process, tokenization also compresses text, which is efficient for storage and for feeding into the language model.

The tokenizer is case sensitive. So the reason that `(T, h)` or `(t, h)` was not the first merge in the example above is because these are not equivalent sequences.

Another important consideration is spaces. In fact, before tokenization, there is another step called **pretokenization**. In this step, a string is segmented according to a set of rules determined by the NLP practioner. The most common pretokenization method is whitespace pretokenization, which entails splitting a text along every whitespace[2]. This means that when the tokenizer learns merges, it will never learn

[2]Another common pretokenization step is to split along all punctuation marks. In this example, the '.' would also be separated, so it would not be possible for the tokenizer to learn a merge `(t, .)`.

merges across a whitespace boundary. While some have proposed treating spaces as their own tokens (Gow-Smith et al., 2022), the most common approach is to mark the spaces on either the first or last character of a word. So, the text in the example above would initially be broken down and represented as:

```
T h e _c a t _i n _t h e _h a t .
```

As a result, for the string "I like to eat apples and bananas", if the tokenizer stores the merge `(a, n)`, it would not affect the `a` and `n` in 'and', because the characters in that word are represented as `_a n d`.

The reason for this is that if spaces are ignored, the sequence can be tokenized, but if you were to decode the tokenization process, it would be impossible to restore spaces in their original place. To illustrate this point, consider the text "London Bridge is falling down". If the tokenizer stores each of these words as individual tokens, the tokenized text would be `[London, Bridge, is, falling, down]`. Each of these tokens is represented by a token id. So the tokenizer outputs a list of integers, e.g. `[15377, 26592, 1204, 17038, 792]`. If the tokenizer did not represent spaces, then if the tokenizer were to convert this back into a string of natural text from the token, the resulting text would be "LondonBridgeisfallingdown".

If spaces are represented in the tokens, e.g. `[London, _Bridge, _is, _falling, _down]`, then when the string of token ids is decoded, the resulting text is "London Bridge is falling down." This represents the notion of lossless compression.

The reason that `London` is not represented as `_London` is because it is the first token in a sequence. Some tokenizers mark the beginning and end of sequence with beginning-of-sequence (BOS) and end-of-sequence (EOS) tokens. EOS and BOS

tokens can be anything, and are specified before training the tokenizer. As one does not want the EOS or BOS token to be a string that occurs naturally in the training data, usually they are strings like "[CLS]" and "[SEP]" or "<|begin_of_text|>" and "<|end_of_text|>",

As tokenization is the first step in language modeling, it affects almost every subsequent step; however, it is not that well understood how tokenization design choices or behaviors affect downstream language model performance. One of the most frequently considered features of a tokenizer is its compression rates. This refers to how much natural language text can be represented in a sequence of tokens. If tokenizer A used five tokens to represent the string "London Bridge is falling down", while tokenizer B used six tokens, tokenizer A would be said to have better compression over the given text.

Compression is important, as the more compression a tokenizer offers, the more information is in each sequence. During training, language models see sequences of a fixed length at every step. So if the maximum sequence length of a model is 512 tokens, each sequence that the model sees during training will be 512 tokens long. For tokenizer A, which offers a higher compression rate, 512 tokens will represent more text than tokenizer B.

1.1.2 Training

Language model training – which is most often called pre-training – seeks to tune the models weights or parameters, which are the values used to compute updated representations and ultimately predict the next word. During pre-training,

the model is presented with a sequence. The model uses its current state to predict each token in the text as a function of the tokens before it in that sequence. For the sequence "London Bridge is falling down", therefore, the language model may predict "Bridge" given "London", then "is" given "London Bridge", and so on. The model sees several sequences (together, called a batch). The accuracy of the model's predictions is rated according its accuracy. After each batch, the model updates its parameters according to the loss function, cross-entropy loss. Cross-entropy loss (Eq. 1.1) is the negative sum over the products of the true token probabilities (p_i) and the log of the predicted token probabilities ($\log(q_i)$). This captures the difference between the true and predicted probabilities.

$$-\sum_i p_i \log(q_i) \tag{1.1}$$

The cross-entropy loss is used to determine the model updates. Higher loss means that the model is making worse predictions, and the model will make more significant updates to its parameters. If the loss is lower, then the model is making better predictions, therefore it does not need to make such significant updates.

During model training, perplexity is used to measure model performance. Perplexity is defined as the exponent of cross-entropy loss (Eq. 1.2). As with loss, lower perplexity corresponds to better predictions, and thus better performance. During training, perplexity is calculated over a held-out test test, which is usually a relatively small sample of text that is representative of the training data. Perplexity of 1 corresponds to perfect prediction. In practice, it is practically impossible to achieve perfect loss on a sufficiently large and diverse dataset, otherwise the model

would likely be over-fit and would generalize very poorly to new data.

$$-b^{\frac{1}{N}\sum_{i}^{N} p_i \log(q_i)} \tag{1.2}$$

Over the course of training, perplexity goes down. At some point, the model converges, and perplexity will plateau. Training should be stopped after the model converges and before the model becomes over-trained.

Returning to the topic of tokenization, if the model sees more information during each sequence, and in each batch as a result, for the same number of training updates, the model will have seen more text. This would likely lead to better performance (Petrov et al., 2023), as seeing more text often leads to a better language model. While many of the details remain unclear, it has been shown that choice of tokenizer can have a significant impact on model performance (Bostrom and Durrett, 2020; Ali et al., 2024, *inter alia*).

Contemporary large language models are trained on extremely large corpora of unlabeled data. One of the most-used datasets is the Colossal Clean Crawled Corpus (C4; Raffel et al., 2020), which is a dataset of text scraped from the web. These datasets tend to be very noisy, usually due to artifacts of the text being taken from the internet. In an analysis of C4, Elazar et al. (2024) found that there is a lot of repetition in these datasets. Some of the most frequent 10-grams in the corpus were '??????????' and '.....', which occur 9 and 7.2 million times in the corpus, respectively.

Some other popular open datasets include the Pile (Gao et al., 2020), RedPajama (Together Computer, 2023), and Dolma (Soldaini et al., 2024) also use C4 or

similar datasets, but also include data from academic articles, code, subtitles from films or YouTube videos, and books.

This first phase of training, which is often called pre-training, creates a base model. Base models are exclusively trained for next-word prediction based on the training corpus as described above. However, most commercially deployed models are not base models, but have additional training steps, called post-training. The most common type of post-training procedure is supervised fine-tuning (SFT). "Supervised" contrasts with the pre-training phase, which is considered unsupervised. Unsupervised refers to unlabelled data, i.e. raw text data without any annotations. SFT usually entails predicting specific labels, answers, or outputs given a specific prompt or question. Many commercial LLMs are "instruction-tuned", i.e. fine-tuned to follow instructions. This is what makes them useful for chatbot and assistant applications. Sometimes this kind of tuning is done with RLHF, which stands for reinforcement learning through human feedback[3]. In this procedure, model outputs are rated by human annotators and then the model is updated in a way that optimizes the model to human preferences.

In this dissertation, I study pre-trained-only models. I do not test or train any models that are fine-tuned or have any post-training. This is the case for several reasons. First, post-training data are scarce for lower-resource languages. A multilingual dataset, called the Aya dataset (Singh et al., 2024), was released recently and contains human-annotated prompt-response pairs for 65 languages. This represents the total breadth of post-training datasets available for lower-resource languages, but

[3]Also, RLAIF, reinforcement learning through AI feedback, where models receive feedback from other models during this tuning step.

this datasets represents many fewer languages than are represented in some of the studies in this dissertation. Therefore, limiting these studies to the languages for which there are post-training datasets would further limit the number of languages represented in these studies. Second, fine-tuning is computationally expensive. The studies in this dissertation were already constrained by academic compute budgets.

Beyond these practical limitations, using post-trained models does not allow researchers to reach as strong of inferences as pre-trained-only models for certain research questions. As the studies in this dissertation focus on the role of pre-training data and tokenization, further fine-tuning models would only obscure the influence of these factors. For many questions relating to Linguistics, Psychology, and Cognitive Science, which are likely to relate to learning from distributional statistics, post-trained models are inappropriate, as the models are no longer learning exclusively from the training corpus, but are further updated using explicit feedback. This is not to say that all questions in these fields can only be addressed with pre-trained-only models.

1.1.3 Architectures

Thus far, I have introduced the key concepts in terms of n-gram language models. These are very rudimentary compared to the state-of-the-art models that dominate the field now. Architecturally, n-gram language models are essentially look-up tables of n-gram frequencies in combination with sampling and smoothing functions. In some sense, contemporary models can be considered to be 2048-grams or 4096-grams, as many language models have context windows of a few thousand

tokens, with 2048 and 4096 being common sizes. As the n-gram lengths are very long, and therefore most possible sequences of 2048 or 4096 tokens will never be seen by the model or will have extremely low probabilities, contemporary models have more complex ways of interpolating across much longer sequences.

The most popular model architecture in the field is the Transformer architecture. Here, architecture refers to the design features of a deep neural network, including size (number of parameters), type of attention mechanisms, and other hyperparameters. Transformer models differ from their predecessors in a few key ways, namely in the attention mechanism (Vaswani et al., 2017), which varies the influence of each token on the processing of every other token. This architectural feature has led to some of the most rapid advances in NLP. First, I will provide an overview of the major components of the Transformer, which are also visualized in Figure 1.1.

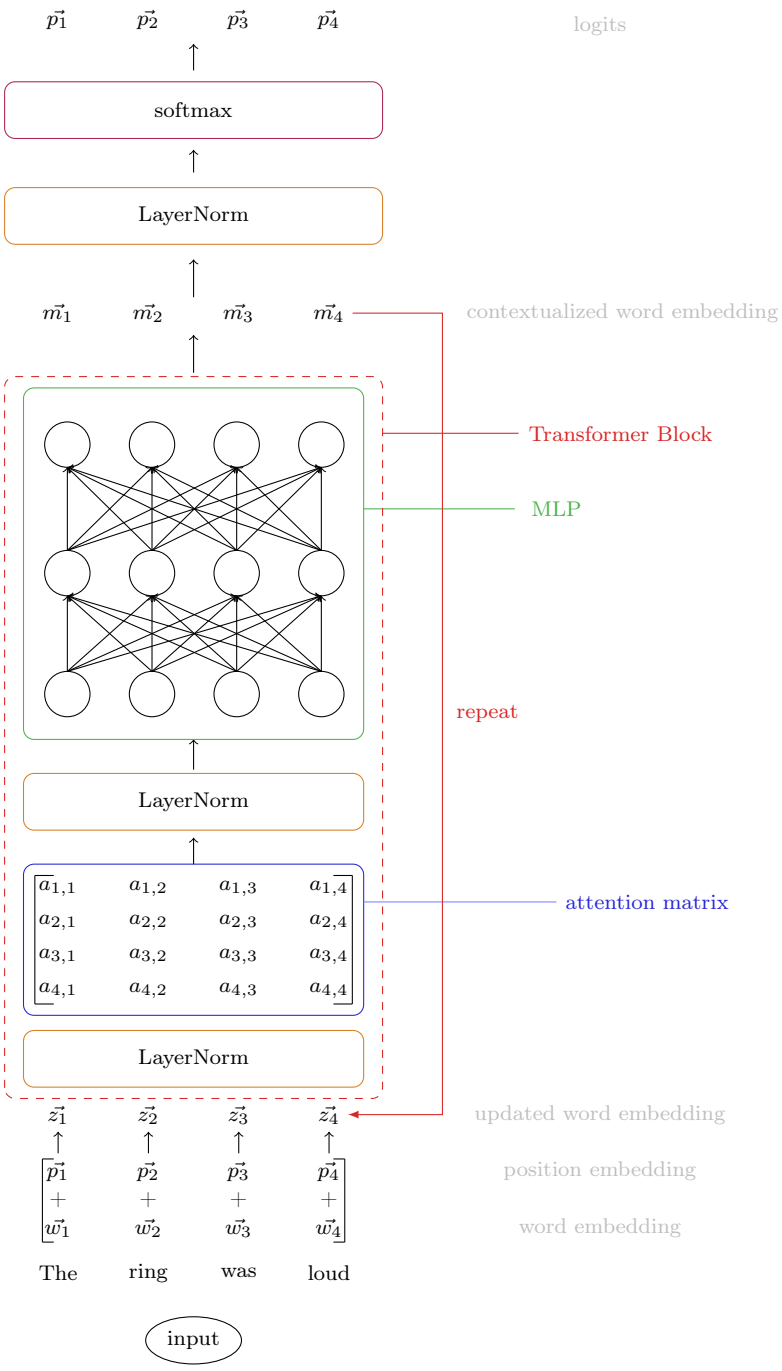

Figure 1.1: Simplified Transformer architecture, based on GPT-2.

The input to a text-only transformer model is a string of language, for example "the ring was loud". The first step, as discussed above is tokenization. For the sake of simplicity, we can imagine the tokenizer tokenizes this string into four tokens: ‘the’, ‘ring’, ‘was’, ‘loud’. The model has a static representation of these tokens, which are called token or word embeddings. This is a vector of numbers, which is used to represent the distributional meaning of a token in the model’s embedding space. Then the token embedding is updated with positional information, which encodes the order of each token in the string. This updated representation is then passed through the transformer blocks. Each Transformer block consists primarily of the attention mechanism and an MLP layer. There are also some additional steps, such as LayerNorm, which normalizes the activations. There are usually many transformer blocks stacked on top of each other. GPT-2 has 12 layers, meaning 12 Transformer blocks stacked on each other. This means that each updated token representation is passed through 12 architecturally equivalent blocks, but which have different weights.

Then at the end there is a final norming layer and then the representations are passed through a softmax function. The input to the softmax is the contextualized token representations, and the output of the softmax is a probability distribution over all the tokens in the model vocabulary. This then allows for the model to use some sampling strategy to predict the next token, e.g. picking the token with the highest probability. These probabilities can also be recorded and studied, as I do in several of the studies in this dissertation.

Token Embeddings

During model training, the token embeddings are part of the learned parameters of the models. Token embeddings can be studied in their own right. From the embeddings, researchers have shown that similar words are represented more closely in embedding space and less similar words are represented more distantly in embedding space (Mikolov, 2013b, *inter alia*). Additionally, there are systematic relationships between word embeddings. Mikolov (2013a) found that after dimensionality reduction with PCA, there were predictable relationships between words with similar relationships, e.g. capital cities and their respective countries (Fig. 1.2).

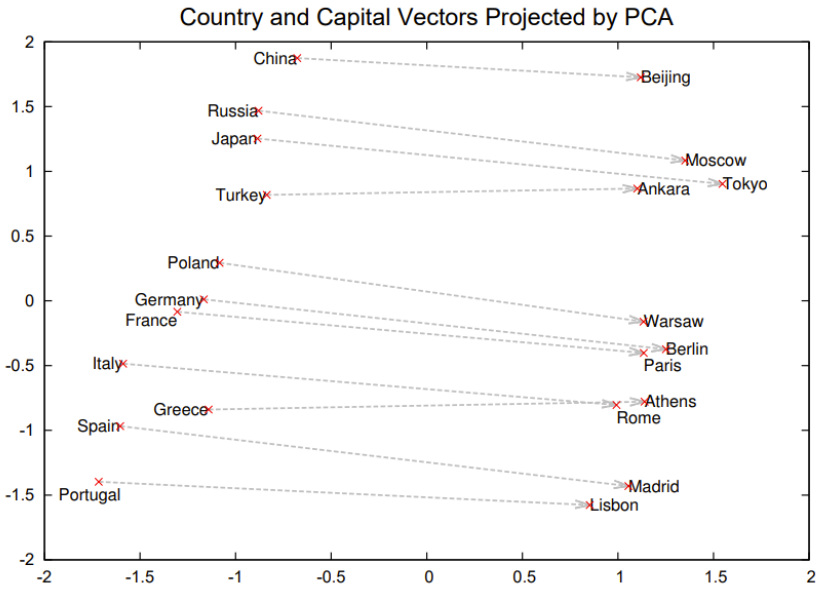

Figure 1.2: (Mikolov, 2013a)

This is true of static embeddings, that is embeddings for words or tokens

irrespective of their contexts. There are also contextualized embeddings, which are the model's representation of a word or token in the context of a specific sentence.

Because each token has a specific number of parameters that the model must learn, equal to the embedding dimensions, e.g. 768, there is a computational limit on the number of vocabulary items in the tokenizer. The number of embedding parameters, therefore scales by a factor of the embedding dimension. The more parameters in the model, the more costly and slow both training and inference are for the model.

Attention

The attention mechanism in a GPT-2-style Transformer model weights the information from each token according to how relevant or informative it is for processing the other tokens. There are broadly two kinds of Transformer models: autoregressive and bidirectional models. These models differ primarily in their attention mechanism. Most contemporary language models are autoregressive, including GPT-2. In autoregressive models, also called causal and unidirectional models, tokens can only attend to tokens earlier in the context. So the first token in a sequence can only attend to itself, the second token can only attend to itself and the previous token, and so on. This is illustrated in Figure 1.3 (left), where grayed out cells in the matrix represent tokens that cannot be attended to. In bidirectional models, also called masked models, however, all tokens can attend to all other tokens in a sequence (Fig. 1.3; right).

As a result, autoregressive and bidirectional models are trained and evaluated

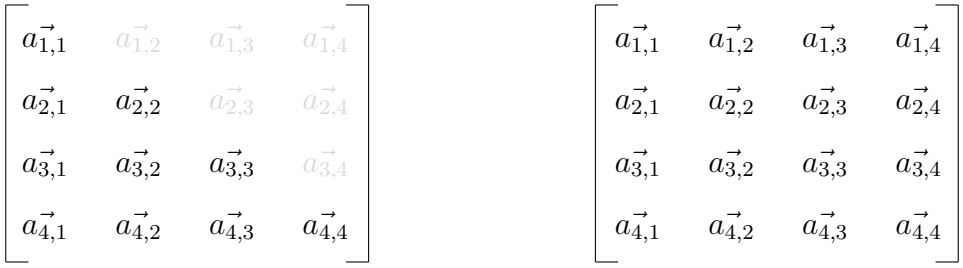

Figure 1.3: Autoregressive (left) vs. Bidirectional (right) Attention. Each column represents an input token and each row represents the tokens each input token is appending to. Each token has an index, which represent its token position, and the position of the token it is attending to. Grayed out tokens are not attended to.

differently. Both models are trained to predict the next token, however autoregressive models predict the next token in a sentence, but bidirectional models are fed an entire sequence with a target token masked out. The model then uses the whole context – preceding and following – to predict the hidden token. These models also make predictions differently. Autoregressive models function well for text generation, by recursively predicting the next token. Given a prompt, the model will predict the next token, then add that token into the context and predict the following token.

These model types can also be evaluated differently. Both models can be used to calculate the probability of a sequence, but other tasks are specific to each types. This point will be discussed in depth in the Section 2.1.5 below.

Open vs. Closed Models In this dissertation, I primarily use open models, with the exception of reporting published evaluations and results. I focus on models to which I have direct access. Some of the most popular models, such at GPT-4 and Claude are accessed exclusively through chat interfaces or Application programming interfaces (APIs). This means that the developer hosts the models and controls

access to the models. As a result, researchers do not have direct access to these models. This is important for at least two reasons. The first reason is replicability. Developers, such as OpenAI, may deprecate a model and no longer provide access to a model to researchers, potentially without notice. GPT-3 is no longer available. Now all work done on GPT-3 is no longer replicable.

The second is that if researchers do not have direct access to the models, it is impossible to verify what the models are. The chat interfaces with these models have built-in prompts (called system prompts), code compilers and other auxiliary tools and functions. But most of the leading developers do not share what additional features have been added to the models. Therefore, it is impossible to say whether users are interfacing with a language model, or with an LLM-equipped software (Trott, 2024b).

One of the barriers between the user and the model is the system prompt. This is an additional prompt used to give instructions to the model about how to respond given the prompt. This can include things like "You are a helpful AI assistant...". But it can also include instructions about tone, style, or format. According to a blog post, OpenAI's newest model o1 has a built-in prompting strategy called chain-of-thought (OpenAI, 2024). Chain-of-thought is a prompting method. Simply by adding the instructions "think step by step", Wei et al. (2022) found that models produced much better outputs, especially for tasks involving reasoning. By simply adding four words to the prompt, they found as much as a 39% increase in performance. The wording of a language model prompt can drastically impact performance, e.g. Lu et al. (2022). For o1, the model does this reasoning step, which is not visible to the

user. Then an output is generated based on the "thinking" step. This means that it is impossible to evaluate o1 without chain-of-thought prompting. These built-in prompts affect model performance, but the user cannot know how much of the model performance is attributable to the model and how much to the prompt (or the specific combination of the two). As a result, evaluating models with system prompts should be understood to be the evaluation of a model-prompt combination. This limits the claims that can be made after evaluating such models.

1.1.4 Language Model Evaluation

An essential part of language model development is to evaluate them in order to measure their performance. One of the primary ways in which language models are evaluated is through benchmarks. Benchmarks are fixed evaluations for language models. Many benchmarks are static datasets with prompts and desired outputs. Models are evaluated on how often they produce the desired output. There are also human-preference benchmarks, such as Chatbot Arena (Chiang et al., 2024), in which human raters compare model outputs. Here I focus on static benchmarks.

I will not focus on application benchmarks, either, including chat, medicine, and legal applications. This falls outside the scope of this dissertation. Finally, I will not discuss benchmarking where LLMs are used to evaluate model outputs. This approach is called LLM-as-judge. There are several limitations to this approach, including position bias, verbosity bias, and bias towards the model's own generations (Zheng et al., 2023). Position bias refers to the observed effect of the order of the compared generations. Models tend to prefer the first generation that is being

compared. Seecond, models tend to rate longer generations as higher-quality, even if the additional material is repetitive or incorrect. Finally, models rate their own generations higher, compared to other models. For these reasons, I will not discuss benchmarks that rely on LLMs as judges.

I will first provide five examples of common benchmarks.

Named Entity Recognition (NER). NER is a task where models identify specific named entities, such as people, organizations, or locations:

(2) a. Mr. $[\text{Robinson}]_{\text{PER}}$ smiled at the teacher.

b. The $[\text{FDA}]_{\text{ORG}}$ announced time travel pills tomorrow.

c. I will arise and go now, and go to $[\text{Innisfree}]_{\text{LOC}}$

(Mayhew et al., 2024)

In this task, each word is labelled as either PER, ORG, LOC, or none of the above. Performance is mesured with F1 score. This task is useful for data annotation, by extracting entities from documents. It also indicates the model's ability to correctly identify important text entities, which is an important component of downstream tasks such as information retrieval.

Grammaticality Judgments with Linguistic Minimal Pairs The representations of grammatical information and the model's ability to generate grammatical sentences can be tested by comparing the probabilities that the model assigns to each sentence in a minimal pair (Linzen et al., 2016). This kind of evaluation is done by comparing the probability of two sentences which differ in only one aspect, making

one grammatical and one ungrammatical, e.g. (3). The model is considered to make the correct judgment if the model assigns a higher probability to the grammatical sentence than to the ungrammatical sentence.

(3) a. The key is on the table.
b. *The key are on the table.
(Linzen et al., 2016)

This principle was used to create evaluation datasets like Corpus of Linguistic Acceptability (CoLA; Warstadt et al., 2019), Benchmark of Linguistic Minimal Pairs for English (BLiMP; Warstadt et al., 2020), and SyntaxGym. All of these datasets exclusively evaluate syntactic knowledge for English. Since their release, there have been equivalent datasets released in other languages. There are Italian, Russian, Hungarian, and Norwegian CoLA-style datasets (Trotta et al., 2021; Mikhailov et al., 2022; Ligeti-Nagy et al., 2024; Jentoft and Samuel, 2023); Dutch, Japanese, and Chinese BLiMP-style datasets (Suijkerbuijk et al., 2024; Someya and Oseki, 2023; Xiang et al., 2021; Song et al., 2022; Liu et al., 2024) ; and Spanish SyntaxGym (Pérez-Mayos et al., 2021). Additionally, there are some smaller, more targeted syntactic evaluation datasets for additional languages, including Basque, Hindi, Swahili, French, German, and Hebrew (Mueller et al., 2020; Kryvosheieva and Levy, 2024). While the language coverage of such benchmarks is growing, there are many high-resource languages that still do not have syntactic knowledge evaluations. Additionally, none of the languages mentioned here have datasets as large or comprehensive as the English datasets.

Natural Language Inference (NLI). The Natural Language Inference (NLI) task evaluates understanding of entailment and contradiction, which is an essential part of language understanding and language processing (Bowman et al., 2015). It is also thought to evaluate logical reasoning (MacCartney and Manning, 2008). In this task, the model is presented with two propositions. Then the model predicts one of three relationship labels: entailment, contraction, and neutral. Example (4), provides examples from the Stanford Natural Language Inference (SNLI) benchmark.

	Proposition 1	Proposition 2	Relationship
a.	An older and younger man smiling.	Two men are smiling and laughing at the cats playing on the floor.	neutral
b.	A man inspects the uniform of a figure in some East Asian country.	The man is sleeping.	contradiction
c.	A soccer game with multiple males playing.	Some men are playing a sport.	entailment

(4)

There is also a multilingual NLI benchmark called XNLI (Conneau et al., 2018), which covers 15 languages.

Question Answering (QA). Question Answering (QA) benchmarks seek to evaluate generalized language understanding in combination with world knowledge and commonsense reasoning. Some QA benchmarks are open-ended, meaning a language model is evaluated by prompting it to generate the answer. One example of an open-ended QA benchmark is the Stanford Question Answering Dataset (SQuAD; Rajpurkar et al., 2016). Multiple choice question-answering (MCQA) tasks are more common. Some examples include Hellaswag (Zellers et al., 2019), PIQA (Bisk et al., 2020), and MMLU (Hendrycks et al., 2021a). Such MCQA tasks are often adopted as part of a small set of benchmarks that are frequently used to compare general-purpose state-of-the-art models. An example from MMLU is shown in (5) below:

(5) What is the embryological origin of the hyoid bone?

(A) The first pharyngeal arch

(B) The first and second pharyngeal arches

(C) The second pharyngeal arch

(D) The second and third pharyngeal arches

(Answer: D)

As models improve and reach ceiling performance on these tasks, they are replaced with harder versions. MMLU, for instance has been superceded by MMLU-Pro (Wang et al., 2024c), MMLU-SR (Wang et al., 2024b), and GPQA (Rein et al., 2024).

Crosslingual version of these datasets exist in some cases. There are several translated SQuAD benchmarks for a variety of languages, including French (d'Hoffschmidt et al., 2020), Russian (Efimov et al., 2020), Bengali (Tahsin Mayeesha et al., 2021), German (Möller et al., 2021), and Persian (Abadani et al., 2021). Belebele is a parallel MCQA benchmark, released for 122 languages (Bandarkar et al., 2023).

Reasoning/Math. Another type of evaluation is mathematical reasoning tasks, such as GSM8k (Cobbe et al., 2021). This benchmark contains mathematical word problems, that require multi-step reasoning in order to solve them. An example of a question from GSM8k is shown below in (6). This is a task that many NLP practitioners believe is important for characterizing general-purpose models.

(6) **Problem:** Tina buys 3 12-packs of soda for a party. Including Tina, 6 people are at the party. Half of the people at the party have 3 sodas each, 2 of the

people have 4, and 1 person has 4. How many sodas are left over when the party is over?

Solution: Tina buys 3 12-packs of soda, for 3*12=36 sodas

6 people attend the party, so half of them is 6/2=3 people

Each of those people drinks 3 sodas, so they drink 3*3=9 sodas

Two people drink 4 sodas, which means they drink 2*4=8 sodas

With one person drinking 5, that brings the total to 5+9+8+3=25 sodas

As Tina started off with 36 sodas, that means there are 36-25=11 sodas left

Final Answer: 11

Challenges in Language Model Evaluation

This list of five evaluation tasks is not exhaustive, but it is sufficiently representative to discuss the challenges faced by researchers seeking to evaluate and compare language models. Here I will introduce some of the challenges with LLM evaluation.

The fallacy of one-size-fits-all evaluation. With the increased pursuit of "general-purpose" AI models (Varoquaux et al., 2024), many people have adopted a one-model-to-rule-them-all approach to LLM development. Language model evaluation is conducted in a way that relies on a handful of tasks, which are used to determine the "best" language model. This leaves only a very narrow view of what makes a good language model. One popular way of comparing language models is with a small number of benchmarks that make up leaderboards like Open LLM Leaderboard[4].

[4] `https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard`

The Open LLM Leaderboard is currently comprised of Instruction-Following Evaluation (IFEval; Zeng et al., 2024), Big Bench Hard (BBH; Suzgun et al., 2023), Math Lvl 5 (Hendrycks et al., 2021b), Graduate-Level Google-Proof Q&A Benchmark (GPQA; Rein et al., 2024), Multistep Soft Reasoning (MUSR; Sprague et al., 2024, and MMLU-Pro (Wang et al., 2024c). IFEval evaluates how well a model follows instructions; BBHU, MathEval, and MUSR all evaluate different types of reasoning and problem solving; and GPQA and MMLU-Pro are QA benchmarks that evaluate commonsense and world knowledge. Therefore, the "best" language models are determined by performance along these dimensions.

This is problematic, as this does not reflect documented uses of LLMs. In an analysis of 1 million user interactions with ChatGPT, Longpre et al. (2024) showed that the majority of user requests were requests for composition – either creative, code, or academic composition. Reasoning and world-knowledge performance are not necessarily predictive of text-generation and composition tasks, including paraphrasing, summarization, and translation. It is important to evaluate models in the context of the desired applications.

Construct validity. Construct validity was proposed in the context of psychological tests (Cronbach and Meehl, 1955) to describe the extent to which an evaluation measures what is intending to be measured. The frameworks developed for psychological research are helpful in identifying challenges to validity in language model research. Poor construct validity is widespread in language model research, which is a result of mismatched theory of a phenomenon and the operationalization of measurement of a particular behavior (Jacobs and Wallach, 2021; Saxon et al., 2024). It has

been shown that when the evaluation metric mirrors the training objective, model performance may be artificially inflated (Saphra et al., 2024). These benchmarks, therefore, may not be accurately showing the potential failure modes of the models in real-world applications. This is compounded, as practitioners seek evaluation of general capabilities, disconnected from concrete applications.

Robustness. There is a significant amount of evidence to show that models are extremely sensitive to context, such that minor prompt variations and formatting can significantly change model performance (Biderman et al., 2024). Syntactic evaluations with benchmarks like BLiMP are sensitive to surrounding context, such that model judgments are more variable when minimal pairs are surrounded by text containing relevant content Sinha et al. (2023). Additionally, the evaluation method significantly impacts performance, even if the evaluation data remains constant. Hu and Levy (2023) showed that performance on minimal pair grammaticality judgments is higher when sampling the sentence probabilities, as opposed to using prompting methods to elicit metalinguistic judgements. This is likely due to the increased task demands of the latter method, which are the auxiliary challenges associated with a particular task (Hu and Frank, 2024).

Goodhart's Law. This law states "when a measure becomes a target, it ceases to be a good measure[5]". Some model evaluations have a very short lifespan (Saxon et al., 2024). If a large industrial developer targets a benchmark, the developer may optimize language model training in order to "beat" a task. Many consider this

[5]Original quote: "Any observed statistical regularity will tend to collapse once pressure is placed upon it for control purposes." (Goodhart, 1975)

to render the task useless. I argue that benchmarks are not useless in this case, because they still may be used to characterize classes of models. Only state-of-the-art or near-state-of-the-art models may beat a certain benchmark, therefore, it may still be used to identify near-state-of-the-art models. One example of this type of benchmark is GPQA (Rein et al., 2024). When the dataset was initially released, the best model the authors tested achieved only somewhat above chance performance (GPT-4; 36%), but a few days before the dataset was presented at a conference for the first time, OpenAI announced that their o1 model achieved 78% accuracy, where domain-expert human accuracy was between 65-80%[6]. It is not the case that language model technologies advanced that significantly between the release and presentation of the dataset (approximately one year), but instead this is evidence that OpenAI targeted the kinds of capabilities being evaluated by GPQA, resulting in drastic improvements in performance. This is a clear example of the dangers in placing too much value in any individual static benchmark.

Reproducibility As in other scientific fields, the threat of a replication crisis looms over NLP. Because the pace of research is very fast in NLP and reviewers prioritize novelty[7], most researchers are not incentivized to replicate prior work. As a result, the vast majority of studies and experimental procedures have never been replicated. This is also due to the lack of reproducibility and transparency (e.g. not releasing experimental code) due to high financial risk, competition, and high cost of exper-

[6]See:`https://www.youtube.com/watch?v=ZANbujPTvOY&t=29s&ab_channel=ConferenceonLanguageModeling`

[7]See reviewing guidelines from ACL and NeurIPS: `https://2023.aclweb.org/blog/review-acl23/` and `https://neurips.cc/Conferences/2020/PaperInformation/ReviewerGuidelines`.

imentation (both computational and monetary). Potential future replication crises may be driven by the fragility of benchmarks as discussed above. This is compounded by the prevalence of the use of closed, proprietary models in academic research. Especially for models, which are available only through APIs, research may become suddenly unreplicable if a model is deprecated. The most notable case of this is the recent deprecation of GPT-3, meaning that thousands of studies are no longer replicable (Biderman et al., 2024).

Other methodological concerns. Finally, there are methodological practices that threaten the reliability of language model evaluations.

Many benchmarks – even widely adopted ones – are of dubious quality, upon close inspection. Reference summaries for standard summarization benchmarks, e.g. XSUM, are of very poor quality (Bommasani et al., 2022). An analysis of MMLU, showed that many questions contained errors or were unclear. In the Virology subset, Gema et al. (2024) found that 57% of the questions had errors. Both the Virology and College Chemistry subsets had very high rates of ground truth errors, and the Professional Law and Formal Logic subsets each had high numbers of questions with multiple correct answers[8].

Another concerning methodological practice in model evaluation is the lack of statistical testing. Despite the fact that common evaluation tools like the LM Evaluation Harness provide standard error (Biderman et al., 2024) and robust evidence that statistical testing is important (Ulmer et al., 2022), many researchers still do

[8]Additionally, some exploration of the datasets have revealed very high proportions of low-quality and incorrect examples in datasets like Hellaswag. One blog post reports as much as 36% of the Hellaswag validation set contains errors (Chen, 2024).

not use statistical testing (Miller, 2024).

Contamination is also a big problem. Contamination refers to the case when the evaluation benchmark is contained within the model training data. It is impossible for researchers to directly test contamination for closed models, because the developers do not release information about the data they train on. There is growing indirect evidence for the severity of benchmark contamination in LLM training data. First Deng et al. (2024) showed that OpenAI and Anthropic models show high accuracy on word-for-word completion of common benchmark tasks. GPT-3.5-turbo, for instance, could predict the exact wording of masked answer choices for an MMLU test set with 57% accuracy.

Models also showed sensitivity to the order in which multiple choice question answers were provided. Gupta et al. (2024) showed that by shuffling the answer choices in MMLU, performance for some models dropped as much as 42.9%. This effect is consistent with answer memorization as a result of contamination. Using these contamination-proof modifications to popular benchmarks, the rankings on popular leaderboards change significantly (Alzahrani et al., 2024), showing how fragile these evaluations are and how great the impact of these minor changes can be.

Many open-data models have taken extensive decontamination measures, e.g. the Pile (Gao et al., 2020), Soldaini et al. (2024). Most of the common evaluation datasets were not found in commonly used pre-training corpora (Elazar et al., 2024).

Discussion

I have discussed many of the challenges that arise from using benchmarks to evaluate language model performance. In addition to these and the scarcity of high-quality evaluations (particularly for medium- and low-resource languages-), not all benchmarks can be used to evaluate all models. One of the primary distinctions that must be considered is architecture, particularly the difference between bidirectional and autoregressive models. Many of the recently developed benchmark formats, such as QA and problem-solving tasks are unsuitable for evaluating bidirectional models. At the same time, tasks that were developed for bidirectional models, such as text classification, named entity recognition (NER), dependency parsing, and part-of-speech tagging are not suitable for evaluating autoregressive models.

Additionally, there is a divide between evaluations for small and large models. Most recent efforts in benchmark development have focused on evaluating state-of-the-art models. These benchmarks often use specialized formats (e.g. multiple-choice questions) and require reasoning capabilities, which only very large models will be able to perform. For example, in the MMLU paper (Hendrycks et al., 2021a), the authors find that 13B parameter models achieve chance performance.

In this dissertation, I often use perplexity as an evaluation even though it is considered a sub-optimal evaluation metric.. This is largely due to practical considerations. Many of the languages I study do not have any evaluations, or have not evaluations for autoregressive models. Perplexity is very useful, as it does not require labeled datasets (Chang et al., 2023a). As discussed above, metrics that mirror the training objective may overestimate model performance; however, I argue that this

is better than having no evaluations at all or excluding languages from NLP research altogether. It has been shown that perplexity is predictive of downstream performance on natural language tasks (Xia et al., 2023); however, it does not necessarily predict performance on higher-order, reasoning-heavy tasks.

One concern about using perplexity as an evaluation metric is that perplexity is not predictive of human-like language processing (Kuribayashi et al., 2021), which is consistent with recent findings that larger models tend to perform worse for predicting some metrics of human language processing, such as reading time (Oh and Schuler, 2023), but not other metrics like N400 (Michaelov and Bergen, 2022a, 2023). In the experiments where I use perplexity, however, I do so to measure language model performance. Perplexity is a metric of accuracy at the next-word prediction tasks, which is the training objective of language models. Therefore, I argue that it is useful as a model evaluation metric, especially where there are otherwise no available evaluations.

1.1.5 Language Models as Model Organisms

In the previous section, I have provided the fundamental background knowledge about language models needed to engage with the empirical work presented in this dissertation. I now discuss the role of language models in linguistic and psycholinguistic research.

Language Models as Models of Language. Language models are the first model organism available to study the structure and usage of natural language, and

their potential contribution to theoretical linguistics has yet to be fully explored. Language models offer a new way to adjudicate between long-standing theoretical disputes, between nativist and usage-based views of language.

The underlying premise behind language models is the distributional hypothesis, which states that words occurring in similar context are likely to have similar meanings (Harris, 1954)[9]. As discussed in Section 1.1.3, language models learn to represent words (actually, tokens) in embedding space, such that similar words are represented more closely together. Furthermore, token representations have systematic relationships in line with systematic meaning correspondences.

Some representatives of the nativist account reject language models as being relevant to the study of language altogether, for at least three primary reasons. First, language models cannot distinguish between possible and impossible languages (Chomsky et al., 2023), that is language models can learn languages that would be impossible for a human to learn in addition to possible and attested human languages. Recent evidence suggests this not to be the case. Kallini et al. (2024) showed that language models learn impossible languages less well that possible languages. This of course depends on having actually identified what distinguishes possible and impossible languages, which is not currently agreed upon.

The second reason is that language models are trained on orders of magnitude more data than humans are exposed to (Piantadosi, 2023). The argument, then, is that language models are not developmentally plausible. The nativist view of language is that humans are able to learn language through relatively little exposure

[9]Also commonly referred to by the quote, "You shall know a word by the company it keeps" (Firth, 1957).

due to innately specified knowledge of the essential properties of language, and because language models depart so much from the "training data" of humans, they cannot be useful for the study of language. There have been recent efforts, however, that show that language models trained on more developmentally plausible training corpora were able to achieve promising results (Mueller and Linzen, 2023; Warstadt et al., 2023), showing that language models do not need that much more data than humans to learn fundamental grammatical structures[10].

Third and finally, is the argument that statistical approaches to language modeling are limited by the reliance on linear string order, which cannot account for hierarchical syntactic structure (Everaert et al., 2015 in Millière, forthcoming). There has been mounting evidence, however, that language models can recover hierarchical syntactic representations for linear inputs (Wilcox et al., 2019). Further evidence has shown that language models are capable of processing and producing text that contains bounded hierarchical phrase structure and recursion (Dąbkowski and Beguš, 2023; Mueller et al., 2022a).

While there has not been widespread adoption of language models in theoretical linguistic research (Millière, forthcoming), there has been a significant amount of empirical work on language acquisition and LLMs. Language models have the capacity to empirically test learnability claims (Elman, 1996 in Millière, forthcoming). Language models have strong implications on acquisition research, as they undermine "virtually every strong claim about innateness of language proposed by generative linguistics" (Piantadosi, 2023), by providing evidence that learning to pro-

[10]See Millière (forthcoming), pp. 30-32 for review and further discussion on this point.

duce grammatical language is possible through statistical learning (Contreras Kallens et al., 2023).

One way in which language models can be very useful as model organisms in acquisition research is in running experiments that manipulate language exposure. Misra and Mahowald (2024) iteratively trained language models, by increasingly removing examples of a particular grammatical structure from the training data to show how language models learn statistically infrequent grammatical structures. Similarly, Patil et al. (2024) removed all examples of target grammatical structures to test whether language models could learn those structures from indirect evidence. Both of these studies show the value of using manipulations on language exposure that are not possible with human subjects.

Language models are not theories of language, but models of language (Müller, 2024). Müller (2024) argues that the place of language models in linguistics is as "subjects that one can feed arbitrary training material and that one can interrogate without them getting tired and without the need of an ethics vote" rather than intrinsic evaluations of language models and their capabilities. Müller (2024) argues the latter is irrelevant for linguistics. It seems unlikely that evaluation of language models will directly lead to progress in linguistic theory. However, through the manipulation and evaluation of language models, linguists can refine hypotheses about language, which may indirectly contribute to linguistics.

There is also a view that the evaluation of language models is "mere exercise in studying engineering artifacts" (Millière, forthcoming; Chomsky et al., 2023). The continual involvement of linguists in the development of language models is also crit-

ical for developing models that better serve the goals of linguists. The separation of linguistics and NLP is mutually harmful to linguistics and NLP, as it means language models that are less linguistically-informed and linguists who benefit less from computational tools and methods. There is a lot for each linguistics and computational sciences to contribute to one another.

Much of the linguistically informed work in NLP involves fundamental concepts from linguistics, for example administering the Wug Test (Weissweiler et al., 2023a) and morphologically aligned tokenization (Hofmann et al., 2021; Jabbar, 2024; Toraman et al., 2023; Batsuren et al., 2024). This research area is promising, however a barrier to the development of linguistically informed work that takes into account more nuance, such as individual variation, is the availability of models which perform well for more than a handful of languages. This is critical for providing the broad linguistic coverage necessary to engage with key questions in linguistics. I will provide discussion of this point in the following section (§1.2).

Language Models as Models of Language Users. Language models may also serve as models of language use, i.e. language comprehension and production. The adoption of language models in psychology and cognitive science research has been more widespread than in linguistics. Given the rich literature on using language models to model human language processing, there is a clear contribution that language models can make (and have already made) to research in those areas. The most pressing concern may be about the limitations of those contributions. There are promising results for using language models to model metrics of human language comprehension like the N400 Michaelov et al., 2022; Michaelov and Bergen, 2022b,

inter alia, which inform our understanding of inform our understanding of human language processing.

Language model surprisal can also be used in conjunction with behavioral experiments to tease apart theoretical alternatives in psycholinguistic experiments. Quinn et al. (2024) used language model surprisal to disentangle the role of predictability from the role of syntactic processing on the influence on inhibitory control in bilingual language production.

One potential limitation of this work is whether language models are cognitively plausible, that is whether they mechanistically resemble the mechanisms of human language processing. In recent work, we compare language models of the dominant Transformer architecture with a newly proposed recurrent architectures (Michaelov et al., 2024). This contributes to an ongoing debate about whether language models can be considered cognitive models (Piantadosi, 2023; Mahowald et al., 2024), as opposed to just computational models. We find that there is no difference in the ability for language model surprisal to predict metrics of human reading comprehension (both reaction time and N400 amplitude). As both the recurrent architectures we evaluated, RWKV (Peng et al., 2023) and Mamba (Gu and Dao, 2024), are relatively new. I hope that as recurrent architectures are further developed, this empirical question can be revisited.

In addition to their role in modeling language processing, language models are also promising tools for augmenting psycholinguistic and neurolinguistic experimentation. Jain et al. (2024) proposed *in silico* testing for neurolinguistic experimentation for hypothesis and evaluating construct validity. This is particularly

valuable in a domain like neuroscience, where experimentation is extremely costly. Some examples of work in this vein include using language models to supplement psycholinguistic norms, in order to help develop balanced psycholinguistic stimuli (Trott, 2024a) and to refine hypotheses for child language acquisition (Misra and Kim, 2024).

1.2 Crosslingual and Multilingual NLP

The architectures and training procedures described in the previous section can in principle be applied for any language or combination of languages. It seems as simple as finding a large enough corpus of raw text data. It is not so simple, however. First, there is not that much data available for some languages. As a result, there have not been attempts to train language models for the vast majority of the world's languages. So there is still not enough crosslingual work to have a comprehensive picture of how language models work differently for different languages. It is also the case that there have not been language models trained on every possible combination of languages. So there is much that is unknown about how languages interact within a single language model. Both of these are exciting research areas, as knowledge of crosslinguistic variation, typology, language acquisition, and more can help develop an understanding of these issues.

I make a distinction between crosslingual and multilingual NLP. By crosslingual NLP, I refer to the role of individual language differences, language-specific data, etc. In this dissertation, I primarily investigate crosslingual NLP questions in

monolingual settings, i.e. using models that are only trained on data from one language, in order to maximally control for confounding factors. I use multilingual NLP to refer to research involving language models trained on data from more than one language. This work must be done in multilingual settings. While it is possible to study crosslinguistic differences in a multilingual setting, I would like to disentangle these points. The first part of the dissertation will focus on crosslingual NLP and the second part will focus on multilingual NLP. Therefore, I do not use the term "multilingual" to refer to non-English, monolingual NLP questions, as is sometimes done.

1.2.1 Crosslingual NLP

It has been argued that true success in NLP means developing language technologies that work equally well for all languages (Choudhury, 2023), which is sometimes referred to as language-independent NLP (Bender, 2011; Khanuja et al., 2023). Most of the developments in NLP focus on improving English performance, but many people operate under the assumption mentioned above, that as long as there is sufficient data and all else being equal[11], that developments for English language technologies can be easily translated to language technologies for other languages. But languages differ in the way they form words; in their writing systems; in the way they encode grammatical information; and in terms of lexicalization, i.e. how languages partition up the meaning space into words. How do these differences interact with

[11]Other factors may include data quality, which should not be assumed to equal at all. Kreutzer et al. (2022) showed that in large, widely-used multilingual datasets, data for lower-resource languages is often low-quality and sometimes is largely unusable.

language technologies like language models?

There has not yet been any empirical evidence that there exist natural languages which language models are incapable of learning; however, neither has there been positive evidence of language models learning all natural languages equally well. Therefore, there is a concern, that the English-centric nature of the field will lead to inherent assumptions and design choices that implicitly benefit English at the expense of other languages (Bender, 2011; Joshi et al., 2020).

This question not just important from the perspective of ideological opposition to inequality, but it also has practical significance. Language models that are language-independent[12] are incredibly valuable for scientists and industrial developers (Bender, 2011). From a commercial perspective, the more languages that are represented in NLP, the larger the AI market. From a more human-centered perspective, if language technologies continue to only work for a small number of languages, speakers of other languages will be left behind in terms of the social and economic benefits of these technologies. The current trajectory of language inequity in NLP will only exacerbate existing social and economic inequalities around the globe.

From the perspective of research and development, there is also the possibility that designing language-inclusive technologies will lead to general technological improvements that will benefit language model performance for all languages.

It remains an open question how these crosslinguistic differences interact with different components of language model architectures and training procedures. This is one of the key questions addressed in this dissertation.

[12]I do not mean as single language model that works equally well for all languages, but an architecture/training recipe that leads to equitable outcomes for all languages.

1.2.2 Multilingual NLP

Researchers have shown that it is possible to train language models on data from more than one language, and with appropriate model and dataset sizes, a single model can process text in many languages, e.g. multilingual BERT (Devlin et al., 2018), XLM-R (Conneau et al., 2020a), and XGLM (Lin et al., 2022). It has also been shown that language models can process languages they have barely seen during training. This phenomenon is called crosslingual transfer.

Crosslingual transfer is a type of transfer learning. Transfer learning is the ability for neural networks to generalize representations learned for one task in order to perform a different task. Crosslingual transfer, therefore, refers to the observed ability for many language models to perform well with very little data in the secondary language. Many researchers have sought to leverage crosslingual transfer to improve performance for languages, especially languages with very little available training data.

The main limitation of crosslingual transfer is the curse of multilinguality, which is when a model reaches its capacity limitations and is not able to accurately represent its knowledge of all the languages it has been trained on (Conneau et al., 2020a; Chang et al., 2023a). So, while crosslingual transfer is a powerful tool for improving performance for low-resource languages, it does not work equally well.

To illustrate the intuition behind this phenomenon, I return to n-grams, as described above. An n-gram language model is much more accurate in contexts similar to the training data. In the toy example above, containing data about books and ponds and benches, the resulting n-gram would perform better for contexts relating

to those concepts, as opposed to novel contexts relating to shops and computers and cities. The probabilities of n-grams containing those out-of-distribution words will be very low, or even zero. By the same logic, language models will will variably generalize to new languages.

In the case of n-grams, the model would assign a probability of 0 to all strings in a language different to the one the model was trained on. In the case of contemporary models, which use sub-word tokenization, models can sometimes rely on shared sub-word strings to work for languages that the models did not see during training. For example, consider the following sentences. They are translation equivalents in English and Spanish.

```
I like to eat apples and bananas.

Me gusta comer manzanas y plátanos.
```

While there are no obvious cognates or shared words between the two sentences, there are shared sequences of characters like 'an', which occurs in both *bananas* and *plátanos*. If a language model has a shared sequence like 'an' as one of the tokens in its vocabulary, this can help the language model make predictions in Spanish, even if it has only been trained on English data, because it has been trained to make predictions for that subword token.

This kind of transfer works best when the languages are more similar. For instance, consider the same sentence pair in English and Mandarin Chinese.

```
I like to eat apples and bananas.

我喜欢吃苹果和香蕉。
```

For this language pair, there are no overlapping sequences. As a result, a language model would be much less likely to easily generalize its training from English to Chinese, or vice versa.

In previous work, my collaborators and I conducted the first controlled study to provide empirical evidence for the factors that influence crosslingual transfer (Chang et al., 2023a). We found that low-resource languages did often show benefits of added multilingual, that is they benefited from crosslingual transfer. But language similarity was the main variable we analyzed that determined the efficacy of transfer. This is consistent with previous work (Conneau et al., 2020b; Gerz et al., 2018a; Winata et al., 2022; Ahuja et al., 2022; Oladipo et al., 2022; Eronen et al., 2023).

This means that crosslingual transfer is limited by data from related languages. For a low-resource language, the most related languages may also be low-resource languages. Therefore, the most similar languages for which there exists large amounts of pre-training data may not be very similar to the target language, limiting the utility of transfer learning. A common approach is to just combine large quantities of English data with the target language data, but we argue that this may have unintended detrimental effects on language model performance in the target language. In a follow-up study, we compared small monolingual models with large, massively multilingual models (i.e. models trained on tens of or even a hundred languages). We found that models over an order of magnitude smaller may outperform larger models due to the curse of multilinguality (Chang et al., 2024).

In the n-gram example above, I illustrate the role of vocabulary overlap. There has been evidence that vocabulary overlap drives successful crosslingual transfer

(Artetxe et al., 2020a; Conneau et al., 2020b; Ahuja et al., 2022). In Chang et al. (2023a), we show that vocabulary overlap was not the most significant explanatory variable, but instead typological similarity was. It seems that shared grammatical encoding strategies are important for successful transfer.

Language models have representations used to store information about a language, which are used for processing and generating text data. Some of those representations are for language-specific information, but others represent information that is shared for different languages (Chang et al., 2022). The shared representations are thought to drive crosslingual transfer. In this dissertation, I seek to characterize those representations and understand how and when they are learned by language models.

1.2.3 The State of NLP Beyond English

One of the greatest challenges to research on crosslingual and multilingual NLP questions is the lack of resources for the vast majority of the world's languages. The vast majority of work is done on English (Joshi et al., 2020; Søgaard, 2022; Blasi et al., 2022). And only a small number of languages, relative to the approximate 7000 languages in the world, are represented at all by language technologies and their applications (Joshi et al., 2020). For the languages that are represented, performance for most languages other than English is very poor (Choudhury, 2023).

Because the problem is so pervasive, it is hard to describe the extent to which NLP for languages other than English lags behind. This is in part due to the tendency for researchers in NLP to not specify the language that they are working

on, especially if that language is English. As a response to this, the #benderrule was proposed, which states "always name the language(s) you're working on" (Bender, 2019). Despite this, at least half of ACL papers do not mention what language or languages the paper is studying, according to an analysis in Ducel et al. (2022). Under the assumption that papers that do not mention any language are likely only working on English, I collected the abstracts of all papers from top ACL venues in the past several years[13]. The number of paper abstracts that mention English is constant at between 15-20% of papers. The number of abstracts that do not mention any language by name or the term 'multilingual', is also constant at around 50-60%. Therefore, I estimate that approximately 80% of work published in the most well-regarded ACL venues is on English.

Work on English is seen as the default (Bender, 2019) and as contributing to the whole field, while work on languages other than English are seen as only contributing to the body of work on those specific languages. This has the side-effect of implicitly devaluing work on other languages, as work on non-English languages is seen as having less impact. This is a negative feedback loop, which will result in the vast majority of NLP work being done English and a small number of other languages.

High-resource languages are generally well-served (Blasi et al., 2022), but even for a high-resource language[14], such as French, which has several dedicated (monolingual) models, e.g. CamemBERT (Martin et al., 2020), and hundreds of millions

[13]Due to changes in the formatting of the ACL anthology, abstracts are not easily available for all venues before 2015. Therefore, I begin my analysis in 2016.

[14]I will use the categorization from (Chang et al., 2023a; Appendix A.7), which lists 28 languages as high-resource languages.

of tokens of training data (Ali and Pyysalo, 2024), performance lags behind English. Many large models, such as the Llama models, are trained on a very high proportion of English data, with relatively small proportions of data from other high- and medium-resource languages like French. While information about the training data is not released, an analysis of the tokenizer has revealed the training data proportions for each language (Hayase et al., 2024). Table 1.1 reports the multilingual MMLU scores for the different sizes of Llama 3.1 models: 8B, 70B, and 405B parameters. In the right-hand column, I report the proportion of training data dedicated to that language, according to (Hayase et al., 2024).

Despite the fact that French only makes up 1.8% of the training data, the model still achieves relatively high accuracy on the French task relative to the English performance (Meta AI, 2024). Still performance is worse for these other high-resource languages relative to English for most model sizes.

Table 1.1: Llama 3.1 multilingual MMLU results and training data proportions by language, as reported in the model card on HuggingFace (Meta AI, 2024).

MMLU (5-shot)	**Llama 3.1 8B**	**Llama 3.1 70B**	**Llama 3.1 405B**	**% Training Data**
English	66.7	79.3	85.2	51.5
French	62.34	79.82	84.66	1.8
Spanish	62.45	80.05	85.08	2.0
German	60.59	79.27	84.36	2.2
Italian	61.63	80.4	85.04	1.0
Portuguese	62.12	80.13	84.95	1.4

For medium-resource languages, like Estonian, the situation is somewhat worse. For large proprietary models, Estonian likely makes up between 0.1-0.6% of the training data from GPT-4, Llama, Mistral, Claude, Gemma (Hayase et al.,

2024). There is at least one existing dedicated Estonian model (Tanvir et al., 2021) according to a survey (Ali and Pyysalo, 2024). However, this is a relatively small (100M parameter) BERT model. Most benchmarks do not cover Estonian, but there are a small number of benchmarks that do evaluate Estonian performance, namely XCOPA (Ponti et al., 2020) and Belebele (Bandarkar et al., 2023). As can be seen in Table 1.2, for the Belebele benchmark, several large models all perform above chance for Estonian, where chance is 25%. Even for GPT 3.5 Turbo, which scores the highest of the models shown here, performance for Estonian is significantly worse than performance for English.

Table 1.2: Results from Bandarkar et al. (2023)

	Zero-Shot		**5-Shot**		
	GPT 3.5 Turbo	**Llama 2 70b Chat**	**Llama 2 70B**	**Llama 1 65B**	**Falcon 40B**
English	87.7	78.8	90.9	82.5	77.2
Estonian	73.1	36.6	53.0	36.3	34.9
Bambara	31.7	29.4	30.3	28.4	29.7

For a low-resource language like Bambara, there are even fewer models and benchmarks. There are no existing monolingual Bambara language models. It is not intentionally seen by models such as Llama, Claude, and GPT-4, meaning that Bambara data may be seen in the pre-training data only by accident. Bambara data is used during pre-training for massively multilingual models like BLOOM (Scao et al., 2022), MADLAD-400 (Kudugunta et al., 2024), Glot-500c (Imani et al., 2023), and XGLM (Lin et al., 2022). There is at least one multilingual model that focuses on African Languages, called Cheetah (Adebara et al., 2024). Bambara has at least one benchmark: Belebele. Table 1.2 shows Belebele performance for a small number of

models. In contrast with English and Estonian, the models are all barely performing above chance.

The vast majority of languages are in an even worse position than Bambara. One of these is Isoko. To the best of my knowledge, there are only about 8MB of data publicly available online for the language, which is equivalent to about 2M tokens. For ultra-low-resource languages like this, there are no dedicated language models. Isoko is one of the languages represented in Madlad-400 (Kudugunta et al., 2024), however there are no existing benchmarks for the language so there are no reported performance metrics.

To put into perspective the extent to which the majority of languages have been left behind, I show a timeline of developments in NLP over the last 60 years (Fig. 1.4). I show the timeline of major model and dataset releases in number of words or tokens. All of the models and datasets are in English. The last time English had only about 2 million tokens collected for NLP research was in the 1960s. The Brown Corpus, which was released in 1964, was the first corpus of English text data with one million words. The data resources for Isoko for NLP are comparable to those for English in the 1960s. While the technologies have improves, such that models can do much more with that amount of data than they could in the 1960s, data is still one of the biggest limiting factors to the development of powerful language models.

So, while progress for English resources has led to the development of corpora in the tens of trillions of tokens, many languages have less than *one millionth* the amount of data.

I have been able to find about 16MB of Bambara data, which is equivalent

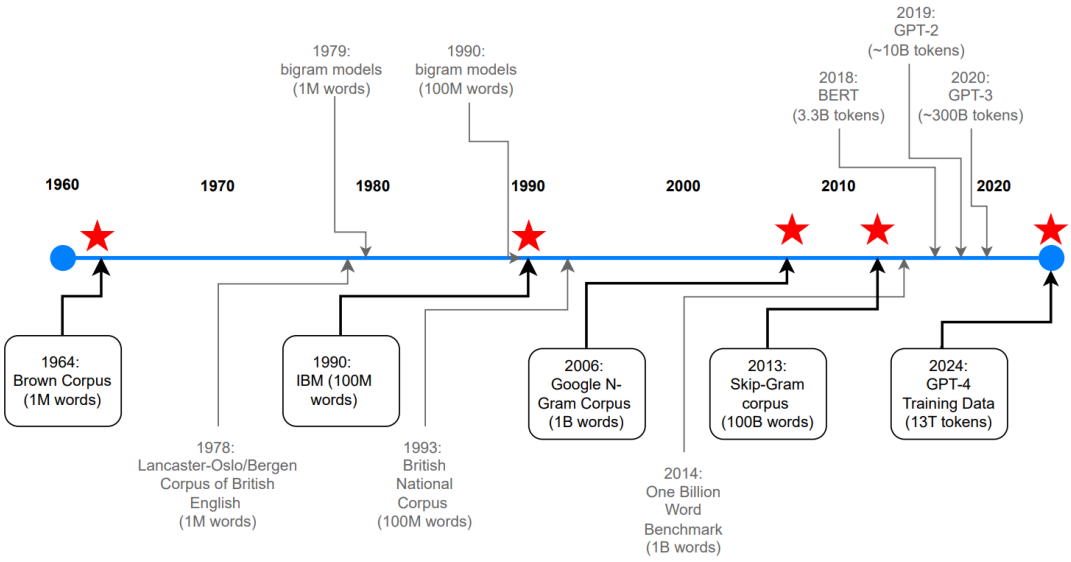

Figure 1.4: Timeline of NLP developments

to about 4.5M tokens. This puts Bambara around the late 1960s in the timeline of English dataset sizes. HPLT is a resource for pre-training data for language models for many languages. Their Estonian corpus is about 20GB, which is equivalent to approximately 3.7B tokens. This makes Estonian a medium-resource language, and this amount of data is equivalent to the late 2000s on the English timeline. This rough comparison begins to illustrate the degree to which different languages are lagging behind English.

Language model performance shows the same kind of patterns. For many medium-resource languages, performance on popular reasoning tasks is still quite low (Table 1.3). In many cases, the language models are performing at or only just above chance, which here is 50%. GPT-3 (Brown et al., 2020), which is now quite far below state-of-the-art performance, achieves 91% zero-shot accuracy on the English

version of this benchmark, COPA (Roemmele et al., 2011). More recently, PaLM (Chowdhery et al., 2023) is reported as achieving 100% on this benchmark.

The Goldfish models (Chang et al., 2024) are small (124M parameter) models, trained on monolingual datasets for 350 target languages. We trained dedicated tokenizers using the same procedure for each language. Both the tokenizer training data and the model training data are scaled according to their byte premium (Arnett et al., 2024a), which will be discussed in Section 2.1. Despite being trained on order of magnitudes less data and being at least one order of magnitude smaller in terms of parameters, the Goldfish models achieve comparable performance on several languages (Table 1.3).

Table 1.3: XCOPA (Ponti et al., 2020) evaluation scores for a selection of models. Higher scores are better and chance performance is 50%.

	Goldfish 124M	**XGLM 4.5B**	**XGLM 7.5B**	**BLOOM 7.1B**	**MaLa-500 10B**
Estonian	0.52	0.552	0.614	0.482	0.53
Haitian Creole	0.554	0.514	0.574	0.508	0.506
Indonesian	0.584	0.668	0.694	0.698	0.61
Italian	0.538	0.614	0.636	0.528	0.606
Quechua	0.512	0.5	0.488	0.508	0.506
Swahili	0.55	0.562	0.6	0.518	0.536
Tamil	0.554	0.554	0.544	0.592	0.562
Thai	0.55	0.554	0.594	0.554	0.55
Turkish	0.562	0.572	0.584	0.512	0.53
Vietnamese	0.574	0.66	0.702	0.708	0.562
Chinese	0.542	0.616	0.638	0.652	0.622

In Chang et al. (2024), we implemented bigram models using the same tokenizers and text data as the Goldfish language models. For XGLM 4.5B, bigram models beat performance in 24 out of the 102 languages it is intentionally trained

on with our bigram models. For BLOOM 7.1B, bigrams beat the model on 20 of 46 languages and bigrams beat MaLa-500 on 11 of the 175 languages overlapping with our language sample and the languages MaLa is trained on.

Both the comparisons to Goldfish and to bigrams show that existing models perform very poorly on many of the languages they are trained on. To give a more qualitative sense of what this means, here is some text generated by an English model trained on only 5MB of text[15], which exhibits similar performance to an ultra-low resource like Isoko:

> This is a new way you are some of some of the same time. If you to start to be good person. No, Thesese are available in the only will not be able took the best time to the best.

So while the best-performing language models, such as GPT-4 and Claude, can handle complex reasoning and problem-solving tasks, deal with data and complex formatting, and follow instructions, for many languages the best language models still are not be able to generate grammatical text without unnatural repetition or typos.

One of the biggest obstacles to language equity in NLP – apart from a lack of resources, both in terms of data and models – is the focus on generalist models. Generalist models are those trained not for a particular task, but to be an assistant or chat bot that can provide generations in many languages, perform well on reasoning-intensive problem solving tasks, and be applicable to a wide range of domains. This one-model-to-rule-them-all approach focuses on developing extremely large models like GPT-4 or Llama 405B. This approach relies on a very large model capacity to allow for crosslingual transfer, despite training on proportionally very little data for

[15] `https://huggingface.co/goldfish-models/eng_latn_5mb`

most languages other than English. In this scenario, preference will always be given to languages with larger populations (Blasi et al., 2022), more economic influence (Bender, 2011), or both. Medium- and low-resource languages will never be served with these models.

It is hard to characterize how poorly models perform for many languages due to the lack of evaluations. Progress in any field is tightly linked with its evaluation paradigm, because metrics of individual success is linked to progress on popular benchmarks (Khanuja et al., 2023). In the absence of benchmarks, researchers are disincentivized from working on technologies for low-resource languages.

1.3 Overview of the Dissertation

Having provided essential assumptions, constructs, and methodologies, as well as the ways that language models can be used as model organisms, I now turn to how these are used in the dissertation.

Part I relates to questions about crosslingual NLP. In this part, I discuss language-specific factors that impact performance and contribute towards linguistic inequities in NLP. I primarily focus on the role of data and tokenization, which are some of the earliest stages of the language model training pipeline. As such, these two factors have a great impact on the downstream components of model training, but they are difficult to study, because of the intervening steps between data and tokenization and the final model performance.

In Chapter 2, I investigate how writing system and word length across lan-

guages affect how dataset sizes are measure in language models and discuss how that impacts language model performance.

In Chapter 3, I investigate how tokenization and morphology interact. I look at whether different tokenization schemes impact performance, specifically whether tokenization which is aligned with the morphological composition of a word, leads to better performance.

In Chapter 4, I compare various factors, namely these two factors (dataset size measurement and how aligned tokenizers are with morphological boundaries), and their effects on crosslinguistic differences in language performance.

In Part II, I use crosslingual structural priming, an experimental paradigm from psycholinguistics to probe shared multilingual representations in language models.

In Chapter 5, I conduct the first experiment showing crosslingual structural priming effects in languages models, showing evidence for the shared representations, which drive crosslingual transfer.

In Chapter 6, I conduct a more controlled follow-up experiment, in which I train bilingual language models to control for data size. I replicate the findings from Chapter 5 and expend on the results, by investigating the training dynamics of language models. I characterize how and when language models acquire shared multilingual representations.

At the end of each Part, I provide discussion. In Chapter 7, I provide conclusions about the work across the whole dissertation.

Part I

Crosslinguistic Differences and Language Models

Chapter 2

Crosslinguistic Data Measurement Inequities in Language Modeling

As was discussed in Chapter 1, the role of data in NLP cannot be overstated. The solution to language inequities – or at least to low performance for low-resource languages is more data. Having equal amounts of data for all languages would be incredibly difficult to achieve, but even if it were achieved, it is unclear how to measure comparable dataset sizes for different languages.

This is an important consideration for at least two stages of language modeling: tokenizer training dataset and model training dataset selection. Datasets are frequently measured in terms of the number of tokens. But before a tokenizer has been trained, data may be measured in terms of words, lines, or in terms of file size (e.g bytes). Word boundaries are often operationalized as whitespaces in NLP. Though there is no consensus on the definition of word in Linguistics, whitespace-

separation is widely agreed to be a problematic implementation. Practically, this is also impossible to implement for languages like Chinese, Japanese, and Korean, which do not use whitespaces. Datasets can be measured by number of lines, but this introduces a high degree of randomness, as the amount of text in each line can vary significantly. Depending on the datasets, lines may be empty or only contain a few characters, whereas for other datasets, a line may contain the text from an entire document. For these reasons, file size in bytes is the most obvious choice.

After a tokenizer has been trained, datasets can selected for model training. At this stage, there are two common choices for dataset measurement: tokens and bytes. The number of tokens is an intuitive choice, as it is linked to important features of the model such as sequence length, i.e. the maximum number of tokens that can be passed into the model at a time and also the number of tokens the model sees at a time during training. Petrov et al. (2023) observed, however that tokenizers introduce disparities between languages, because they offer different compression rates for different languages and domains. This is critical, as Petrov et al. (2023) argue this leads to worse model performance, because the models see less information over the course of training. Higher tokenization costs, especially for large commercial models, also leads to increase cost on top of worse performance (Ahia et al., 2023). Most model providers charge by the token, for both input and output, therefore variable token premiums between languages means variable costs. This most disadvantages users that prompt the model in low-resource languages or languages written with non-Latin script.

For a given string, the number of tokens needed to represent that string de-

pends on the tokenizer. If the tokenizer uses fewer tokens to represent the string, the string will be more compressed. When passed through the model, a sequence with more compression will allow the model to see more text in a fixed sequence. Conversely, if a tokenizer compresses a string less, the model will see less information per sequence. For the same dataset, its size in number of tokens, therefore is dependent on the tokenizer.

Therefore, measuring dataset size in number of tokens means that the metric is variable and depends on the specific tokenizer. Additionally, there will be disparities, such that languages with higher token premiums will be disproportionately underrepresented.

If dataset size is not measured in tokens, then it needs to be measured in bytes. This is the unit of measurement I discuss in the paper below. We show that some languages need more bytes to represent content-matched text. To better contextualize this result, I will first provide some background information about bytes and encoding standards.

Bytes are the unit of measurement of digital information. A byte is comprised of eight bits (1s and 0s). The encoding of text determines how it is stored, and therefore how text file sizes are determined. The most common encoding standard is UTF-8 (Unicode Transformation Format – 8). This encoding standard is designed to be able to render text in any writing system. UTF-8 represents all characters with one- to four-byte code units. Therefore, some characters take more bytes than others. Each byte represents the amount of storage space it takes to represent that character.

In UTF-8 encoding, there are four types of characters: one-byte, two-byte, three-byte, and four-byte characters. There are 256 one-byte characters, because there are 256 unique 8-bit strings, where each bit is either `0` or `1` ($2^8 = 256$). There are 2048 two-byte characters. The first byte is always in a subset of 32 one-byte strings and the second byte is in a subset of 64 possible one-byte strings. 2048 represents every possible combination of those bytes. By the same principle, there are 65,536 possible three-byte characters, and 2,097,152 possible four-byte characters, but not all of them are used. It is worth noting that there are more possible characters represented by the higher-number byte strings. Only a very small number of characters are represented by a single byte.

As a result, different writing systems are not represented equitably by UTF-8. Languages that use Latin script are more likely to have one-byte characters. This is likely primarily due to the fact that when it was introduced, it was made to be backwards compatible with ASCII (American Standard Code for Information Interchange). This system was developed primarily to represent English characters and other frequently used symbols, e.g. punctuation. There are only 128[1] ASCII characters, and all Latin characters without diacritics are among them. When UTF-8 was created, ASCII characters were adopted into the 1-byte group. Initially, not so many writing systems were represented in UTF-8. As more writing systems get represented, they are added to the higher-byte groups. The writing systems that were added earlier tend to be represented with fewer bytes. For instance, many Cyrillic characters are represented with two bytes. Many Chinese characters are represented

[1]ASCII uses 7-bit strings, so there are only 128 unit 7-bit strings.

with two bytes, but some of the less frequent ones are represented with three bytes. Arabic, Devanagari, Thai, Tibetan, and Georgian all have characters primarily represented with three bytes. These characters may not represent diacritics, which might separately be represented as three-byte characters.

Based on this observation, we hypothesized that this could be exacerbating the relative lack of data for non-English – and especially low-resource – languages. We introduce the term *byte premium*, to refer to this difference in the number of bytes needed to encode a given amount of information across languages. Low-resource languages are more likely to use writing systems that are represented with two- and three-byte characters. So languages which might already not have a lot of text data available might have effectively less data due to these disparities in dataset size measurement.

Byte Premiums and Performance

In this chapter (Section 2.1.7), we analyze whether the byte premiums help explain differences in performance between language for existing massively multilingual models. The results showed a numerical, but not significant, difference in the predictive power of byte-premium-scaled data proportions on language model performance when compared to raw data proportions. These results are inconclusive, because of the lack of statistical power. The analysis is also limited, because we did not control the amount of data from each language in each model. There is likely a confound between the amount of data each model is trained on for each language and the byte premium, as lower-resource languages are more likely to have higher

byte premiums.

To follow up on these results, I compare the relationship between byte premium and performance for two sets of models. In this analysis, I compare monolingual model performance, where each model is trained on the same amount of data. In the first set of models, the training data is not scaled according to byte premium, but in the second set of models the data is byte-premium scaled.

The first set of models were introduced in Chang et al. (2023a). In this paper, we trained up to 12 models per language, for 252 languages. We varied two factors: model size and dataset size. There were three model sizes: 8.7M (tiny), 19.8M (mini), and 45.8M (small) parameters. There were four dataset sizes: 1M, 10M, 100M, and 1B tokens. We measure model performance as perplexity over a held out portion of data. I use the byte coefficients from the paper below.

I fit two linear mixed effects models. The full model predicts perplexity using the byte premium as a fixed effect with random intercepts for model size and dataset size. I fit a reduced model, which is the same, but without the fixed effect of byte premium. I compare the two models with an ANOVA and find that the full model explains significantly more variance than the reduced model ($\chi^2(1) = 25.149, p < 0.001$). This suggests that byte premiums explain model performance above and beyond model size and dataset size. The full model shows a positive correlation between byte premium and perplexity (see Fig. 2.1), where a larger byte premium is associated with higher (i.e. worse) perplexity.

These results support our hypothesis that byte premiums may lead to language models being trained on effectively less data for languages with higher byte

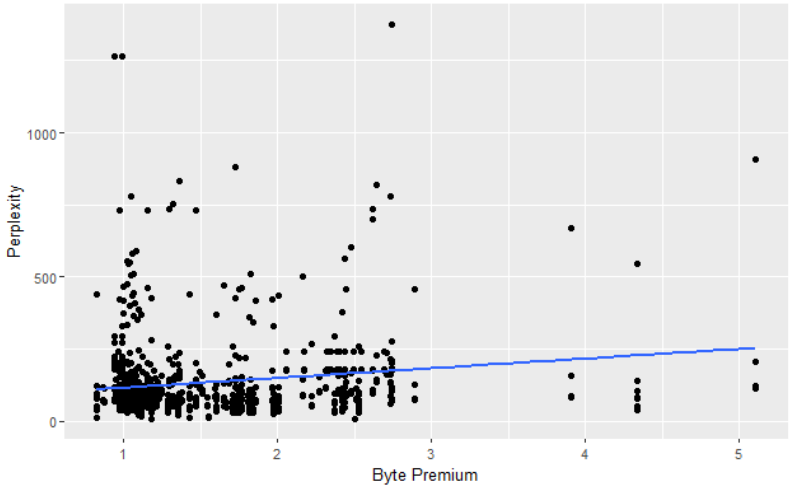

Figure 2.1: The relationship between byte premium and perplexity for the models from Chang et al. (2023a). Each dot represents a monolingual model. There are different models according to dataset size and model size. The blue line represents the line of best fit between the two variables.

premiums.

We also trained a set of models where the training data was byte-premium-scaled, the Goldfish models (Chang et al., 2024). Each model is trained on the byte-premium-scaled equivalent dataset sizes, e.g. 1GB of text. For English, the training dataset size is 1GB because we are using English as the reference point for the byte premiums. For a language with a byte premium of 1.4 relative to English, its training dataset size would be 1.4GB.

In this set, we cover 350 languages. Each language has up to 5 models, according to how much training data was available for that language. The dataset sizes were 5mb, 10mb, 100mb, 1000mb (1GB). So if a language had 100mb of text available, we trained a model each on 5mb, 10mb, and 100mb of text.

In the same way, I analyze perplexity on a held-out dataset and its relationship to byte premiums. I fit a full and reduced linear mixed effects model and found that for the Goldfish models, byte premium did not explain additional variance beyond dataset size ($\chi^2(1) = 0.521, p = 0.470$). The data are plotted in Figure 2.2.

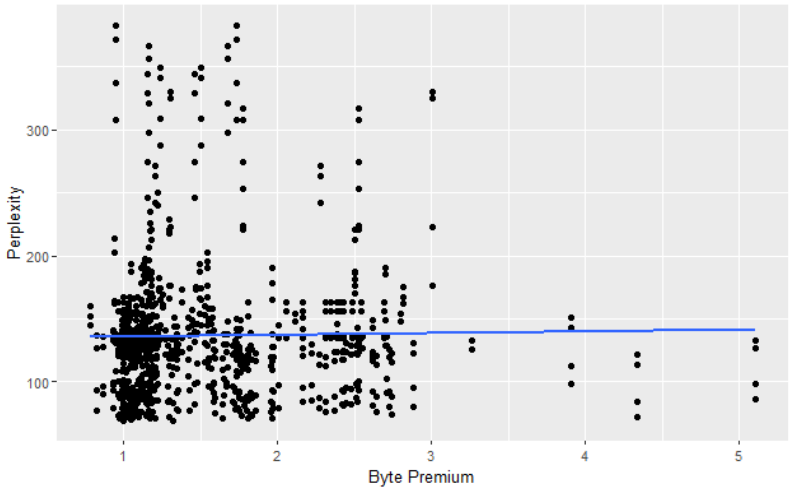

Figure 2.2: The relationship between byte premium and perplexity for the models from Chang et al. (2024). Each dot represents a monolingual model. There are different models according to dataset size and model size. The blue line represents the line of best fit between the two variables.

The results from the Goldfish models suggest that after controlling for byte premium, there is no longer a difference in performance according to a language's byte premium.

Together, the analyses of performance of these two sets of language models suggests that byte premiums explain some of the variance in performance between language models and suggests that taking byte premiums into account can help reduce language inequities in language model performance.

Byte Premiums and Tokenizer-Free Language Modeling

There is another area where byte premiums have serious implications. There is a line of research in NLP which investigates whether language models can use characters or bytes as the unit of language modeling, as opposed to words and subwords. This is usually referred to as "tokenizer-free" language modeling (Choe et al., 2019)[2].

This method is not very popular, because it creates sequences which are very long. This makes the models extremely expensive to train and run inference on (Deiseroth et al., 2024). This area has not been very active until recently, when a new architecture was proposed, called Mamba (Gu and Dao, 2024). When it was released, this architecture was reported to handle long sequence lengths very well. Some researchers have shown that it could be promising for tokenizer-free language modeling (Wang et al., 2024a). One concern is that this method could lead to some of the same inequalities observed by Petrov et al. (2023) for subwords. If NLP practitioners develop a way to make byte-based tokenization work well for English, it may not transfer well to languages with high byte premiums, because the sequences will contain less information than they do for languages like English.

Since the release of our paper, Limisiewicz et al. (2024) proposed a method for addressing variable byte premiums for byte-based tokenization. They propose a new tokenization method called MYTE, which re-maps unused single-byte characters into language-specific morpheme representations. They do this using supervised morpho-

[2]This approach is not really tokenizer-free, as it still involves a text-segmenting step with a fixed vocabulary (either the set of all bytes or a pre-defined list of characters).

logical parsers and determining the highest-frequency morphemes for each language. They show that this method significantly reduces inequities in token premiums while also improving performance for every language they evaluated. While this method is limited to languages for which there exist morphological parsers, this demonstrates the impact of taking into account byte premiums.

2.1 A Bit of a Problem: Measurement Disparities in Dataset Sizes Across Languages

Abstract

How should text dataset sizes be compared across languages? Even for content-matched (parallel) corpora, UTF-8 encoded text can require a dramatically different number of bytes for different languages. In our work, we define the byte premium between two languages as the ratio of bytes used to encode content-matched text in those languages. We compute byte premiums for 1155 languages, and we use linear regressions to estimate byte premiums for other languages. We release a tool to obtain byte premiums for any two languages, enabling comparisons of dataset sizes across languages for more equitable multilingual model development and data practices.

2.1.1 Introduction

Large language datasets serve as the foundation for modern natural language technologies. However, an often ignored question is how to compare dataset sizes across languages. For standard multilingual language models such as XLM-R, BLOOM, and XGLM, dataset sizes are reported in bytes (Conneau et al., 2020a; Scao et al., 2022; Lin et al., 2022).[3] However, content-matched (i.e. parallel) text in two languages does not generally have the same size in bytes, with some languages taking over 5× as many bytes as others (§2.1.3).

Here, we compute **byte premiums** (cf. tokenization premiums in Petrov et al., 2023), the ratios of bytes taken to encode text in 1155 different languages. We find that these byte premiums are highly correlated across datasets. We fit linear regressions to estimate byte premiums for languages not included in our parallel datasets, and we release a simple Python tool to retrieve or predict the byte premium between any two languages.[4] Our work enables comparisons of dataset sizes across languages, with implications for equitable multilingual model development and resource distribution.

2.1.2 Related Work

Using UTF-8 encoding, which is by far the most widespread text encoding (Davis, 2012), characters take between one and four bytes to encode (Unicode Consortium, 2022). Numbers and Latin characters without diacritics are one byte, and

[3]Dataset sizes are also often reported in tokens, which depend on model-specific tokenizers and which exhibit similar cross-language disparities to bytes (Petrov et al., 2023).

[4]`https://github.com/catherinearnett/byte-premium-tool`

all non-Latin scripts use two or more bytes per character. This alone introduces a disparity in measured dataset sizes in bytes (Costa-jussà et al., 2017), but it must be balanced with the fact that different scripts encode different amounts of "information" per character. For example, Mandarin has high UTF-8 bytes-per-character, but it generally requires fewer characters than Latin-script languages to encode the same content. To account for this trade-off, previous work has used parallel text, finding that byte-level tokenizers encode parallel text in some languages using more "tokens" (bytes) than others ("tokenization premiums"; Petrov et al., 2023). We tie these results to dataset storage and training dataset size measurement, we compute the byte premium for 1155 languages, and we present a method to predict the byte premium for novel languages. All our results use UTF-8 encoded text.

2.1.3 Computing Byte Premiums

In this section, we calculate the **byte premium** $\text{BP}_{A/B}$ for different language pairs, which we define as the ratio of bytes taken to encode a comparable amount of information in language A relative to language B. For example, if A on average takes twice as many UTF-8 bytes to encode the same information (parallel text) as B, then $\text{BP}_{A/B}$ would be 2.0. These byte premiums are useful when measuring "how much" content is in each language in a corpus. In multi-parallel corpora, we note that the byte premiums must satisfy:

$$\text{BP}_{A/B} = \frac{\text{Bytes}_A}{\text{Bytes}_C} * \frac{\text{Bytes}_C}{\text{Bytes}_B} = \frac{\text{BP}_{A/C}}{\text{BP}_{B/C}} \tag{2.1}$$

This implies that if the byte premium is known for every language relative to some language C, then all pairwise byte premiums are determined. Thus, we only calculate a single byte premium $\mathbf{BP}_A = \mathrm{BP}_{A/C}$ per language, all relative to reference language C. We use C = English as our reference language, but using any other reference language C_0 would simply multiply all our byte premiums by a constant BP_{C/C_0}. In later sections, we refer to byte premiums relative to English unless otherwise noted. In contrast to Petrov et al. (2023), calculating a single byte premium per language allows byte premiums to be used for multilingual corpora beyond just pairwise corpora.[5]

NLLB

Computing byte premiums requires parallel corpora in the desired languages. We first use NLLB (Costa-jussà et al., 2022), a dataset of pairwise parallel text segments in 188 languages. We sample the first 100K parallel text segments for each language pair (A, B), and we compute $\mathrm{BP}_{A/B}$ as the mean ratio of bytes used in language A versus B, averaged over all segments. This produces a byte premium value for every language pair.

To fit a single byte premium $\mathrm{BP}_A = \mathrm{BP}_{A/C}$ for each language relative to a reference language C (in our case English), we minimize the mean squared error of $\mathrm{BP}_A/\mathrm{BP}_B$ relative to the ground truth $\mathrm{BP}_{A/B}$ (Equation 2.1) over all language pairs (A, B). In other words, we fit 188 byte premium values (one per language) based on

[5]For example, if Equation 2.1 does not hold, then English-Mandarin and Arabic-Mandarin byte premiums could produce conflicting comparable dataset sizes when adding Mandarin data to an English+Arabic corpus.

all 2656 pairwise byte premium values. Fitting these single byte premiums ensures that Equation 2.1 holds for all pairs.

Byte premiums computed from NLLB are reported in Appendix Table A.1. For example, Burmese has byte premium 5.10, so on average it takes 5.10× as many UTF-8 bytes to encode text in Burmese versus English. These byte premiums are consistent when computed from different subsets of the NLLB corpus, with Pearson's $r > 0.999$ for byte premiums computed from ten disjoint subsets of 10% of the NLLB corpus. Notably, byte premiums computed from only 100 lines of text per language pair correlate with the byte premiums computed from the full NLLB dataset with Pearson's $r = 0.955$, indicating that byte premiums can be computed from fairly small parallel corpora.

Other Parallel Corpora

For comparison, we also compute byte premiums from three multi-parallel corpora: FLORES-200 (Costa-jussà et al., 2022; 204 languages), the Bible (eBible, 2023; 1027 languages), and the Universal Declaration of Human Rights (Vatanen et al., 2010; UDHR; 241 languages). For each language A in each dataset, we compute $\text{BP}_A = \text{Bytes}_A/\text{Bytes}_C$ relative to reference language C = English. Because each dataset is comprised of parallel text across all included languages, these byte premiums already satisfy Equation 2.1.

Computed byte premiums are highly correlated between NLLB, FLORES, and the Bible (Table 2.1; Pearson's $r > 0.90$), suggesting that byte premiums are fairly consistent across datasets. We posit that lower correlations with UDHR byte

Table 2.1: Pearson correlations between byte premiums calculated from different datasets. Correlations are high between NLLB, FLORES, and the Bible.

	NLLB	FLORES	Bible	UDHR
FLORES	**0.919**		**0.938**	0.737
Bible	**0.921**	**0.938**		0.177
UDHR	0.592	0.737	0.177	

premiums may be because the UDHR corpora are much shorter (roughly twenty total lines of text) and potentially more domain-specific than the other corpora. For this reason, we do not use UDHR in later sections.

Byte Premiums After Compression

Interestingly, we find that byte premiums persist after compression with the common compression algorithm `gzip` (at maximum compression level 9). Byte premiums after compression by `gzip`, compared to those before compression, are plotted in Figure 2.3. When byte premiums are computed from the compressed FLORES corpora, they correlate strongly with the uncompressed byte premiums (Pearson's $r = 0.890$).

However, the scale of variation across languages reduces substantially after compression; for example, uncompressed byte premiums of 4.0 are roughly analogous to compressed byte premiums of 1.7 (i.e. compressed data in that language takes only 1.7× as many bytes as the reference language rather than 4.0× as many bytes). This suggests that standard compression algorithms reduce but do not fully alleviate disparities in dataset storage sizes across languages.

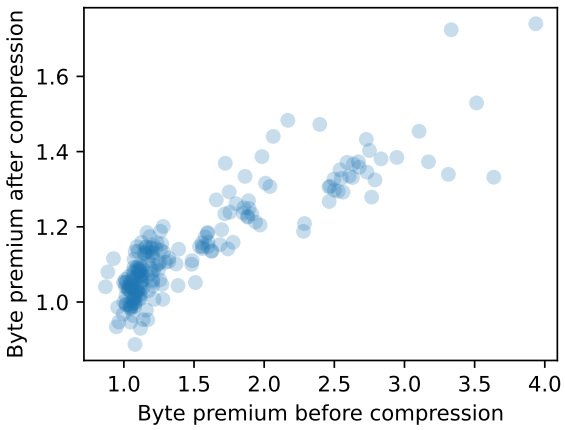

Figure 2.3: Byte premiums before and after compression by `gzip`. Each point is a language's byte premium relative to English.

2.1.4 Predicting Novel Byte Premiums

In many cases, we may want to compute the byte premium for a language A outside of our existing datasets. If a single parallel text is available from A to any language B in our datasets, then the byte premium can easily be calculated as (using reference language C as before):

$$\text{BP}_A = \frac{\text{Bytes}_A}{\text{Bytes}_C} = \frac{\text{Bytes}_A}{\text{Bytes}_B} * \text{BP}_B \tag{2.2}$$

However, there may be cases where no parallel text is available for language A. In this scenario, we can break the byte premium into (1) the mean bytes-per-character in A and C, and (2) the ratio of characters needed to express the same information in A and C (the "length ratio"):

$$\text{BP}_A = \frac{\text{Bytes}_A}{\text{Bytes}_C} = \frac{\text{Bytes}_A}{\text{Chars}_A} * \frac{\text{Chars}_A}{\text{Chars}_C} * \frac{\text{Chars}_C}{\text{Bytes}_C} \tag{2.3}$$

The bytes-per-character ratio for A can be calculated with only monolingual text in A. We find that this ratio is highly consistent regardless of the dataset used. The computed bytes-per-character ratios correlate strongly (Pearson's $r > 0.99$) when calculated from any of NLLB, the Bible, or FLORES with 20, 200, or 2000 lines of text. Given the consistency of these bytes-per-character ratios, we find it efficient to break byte premiums down as in Equation 2.3 such that we only need to predict the length ratio between languages.

Predicting Length Ratios

We use linear regressions including language family, script (writing system), script type (e.g. alphabet vs. logography), and entropy over characters to predict the length ratio $\text{Chars}_A/\text{Chars}_C$ for a language A relative to the reference language $C = \text{English}$. From the predicted length ratio, we can use Equation 2.3 to calculate the predicted byte premium for language A. Our results use length ratios, bytes-per-character ratios, and character entropies computed from NLLB, FLORES, or the Bible when available, in order of decreasing priority.[6]

Language Family We predict that typological features (e.g. inflection patterns or morpho-syntactic distinctions) may drive differences in length ratios. Languages that are in the same language family are more likely to share typological features due to their shared historical origin (Moravcsik, 2012).

[6] As with byte premiums, the choice of reference language C only multiplies all length ratios by a constant. NLLB length ratios are computed in the same way as byte premiums, but using characters instead of bytes. We obtain similar regression results using length ratios, bytes-per-character ratios, and character entropies computed from NLLB, FLORES, or the Bible (Appendix **??**).

Script and Script Type Some writing systems may encode higher information content per character than others (e.g. Chinese characters; Perfetti and Liu, 2005), which leads to low length ratios, because the same content takes fewer characters to write. We separate scripts into four script types (alphabet, abjad, abugida, and logography), and we use script type as a predictor for length ratio.

- **Alphabets** are writing systems where each segment (either consonant or vowel) is represented by a symbol (Daniels, 1990). Latin script is one of the most widely used alphabets. Other alphabets include Greek, Cyrillic, and Mkhedruli (Georgian).

- **Abjads** are writing systems which represent each consonant with a symbol (Daniels, 1990), but vowels are often not represented. Arabic and Hebrew are written with abjads, for example.

- **Abugidas**, also sometimes referred to as *neosyllabaries*, represent consonant-vowel sequences, often with vowel notation secondary to consonant notation (Daniels, 1990). Examples of abugidas include Devanagari (e.g. Hindi), Ge'ez (e.g. Amharic), and Canadian syllabics (e.g. Ojibwe).

- **Logographies** are different from alphabets, abjads, and abugidas in that they represent semantic information as well as phonetic information. Chinese characters are the only logography that remains in use. The majority of Chinese characters are composed of one semantic component and one phonetic component (Williams and Bever, 2010). A relatively small number of characters are

also pictographs or ideographs, representing only semantic information (Ding et al., 2004).

In addition to script type, we also consider the specific script as a nested predictor (e.g. Latin vs. Cyrillic).

Character Entropy It has been proposed that languages with fewer phonemes (contrastive sounds) in their inventories have longer words, because it requires more sounds per word to generate the number of contrastive sound sequences necessary to communicate (Nettle, 1995).[7] Using the same logic, we predict that a language that tends to use fewer unique characters will require longer character sequences to express information (a high length ratio). We operationalize the number of unique characters in a language as the entropy over the character probability distribution in that language. A higher entropy indicates either a more even distribution over characters or a distribution over more characters. Similar to bytes-per-character ratios (§2.1.4), the entropy over characters is highly stable across datasets, even computed from as few as 20 lines of text (Pearson's $r > 0.90$ for the same datasets as §2.1.4).

We fit linear regressions to predict length ratios from three different subsets of our predictors. This allows us to predict novel byte premiums depending on the available information about the novel language. We consider the following three subsets: (I) character entropy, language family, script, and script type, (II) character

[7]We also measure the number of phonemes per language (PHOIBLE; Moran et al., 2014), but it does not help predict length ratios ($R^2 = 0.002$). Therefore we do not include it in our linear regressions.

Table 2.2: RMSEs when predicting byte premiums using different regressions, for languages with common and uncommon scripts.

	Regression		
	I	II	III
Scripts with count ≥ 5	**0.261**	0.288	0.290
Scripts with count < 5	0.770	0.739	**0.589**

entropy, script, and script type, and (III) character entropy and script type. The predicted length ratios can be used to predict byte premiums using Equation 2.3.

2.1.5 Evaluating Byte Premium Predictions

We validate the byte premium predictions from our linear regressions by looping through languages with known byte premiums (from NLLB, FLORES, or the Bible, in that order of priority), evaluating the byte premium prediction for that language when holding it out from regression fitting.[8] We report the root mean squared error (RMSE) for the three linear regressions described in the previous section (I, II, and III). We compute separate RMSEs for (1) languages whose script is shared by less than five languages in our datasets, and (2) languages whose script is shared by five or more languages in our datasets. Languages whose script is uncommon may have more poorly fitted script coefficients (and potentially language family coefficients), so we might expect them to exhibit larger byte premium prediction errors.

Results are reported in Table 2.2. For languages with common scripts (scripts with count ≥ 5), the regressions improve as predictors are added (III, II, then I). For

[8]To prevent skew of regression coefficients, we clip byte premiums to a maximum of 4.0 (three languages; Appendix A.1).

these languages, RMSEs reach 0.261, indicating that the predicted byte premiums are on average approximately 0.261 away from the true byte premiums.

In Table 2.3, we report validation RMSEs for each regression when computing character entropies and bytes-per-character ratios from different datasets. Within each dataset, we separate the languages for which there are less than five other languages with the same script in the dataset from those which have five or more languages with the same script in the dataset. RMSE results are similar regardless of the dataset used to compute character entropies and bytes-per-character ratios.

Table 2.3: RMSEs when predicting byte premiums using different datasets to compute character entropies and bytes-per-character ratios. Results are separated into common and uncommon scripts.

		Regression		
		I	II	III
NLLB	Script ct. ≥ 5	0.201	0.244	0.240
	Script ct. < 5	0.700	0.744	0.637
Flores (20 lines)	Script ct. ≥ 5	0.203	0.246	0.250
	Script ct. < 5	0.682	0.557	0.538
Flores (200)	Script ct. ≥ 5	0.204	0.252	0.254
	Script ct. < 5	0.702	0.615	0.544
Flores (2000)	Script ct. ≥ 5	0.206	0.266	0.271
	Script ct. < 5	0.703	0.647	0.558
Bible (4 books)	Script ct. ≥ 5	0.272	0.294	0.298
	Script ct. < 5	0.766	0.680	0.577
Bible (1 book)	Script ct. ≥ 5	0.271	0.293	0.297
	Script ct. < 5	0.760	0.672	0.566

As expected, we also find that languages with uncommon scripts (scripts with count < 5) have higher errors in their predicted byte premiums, indicating that their script and family coefficients are poorly fitted. Likely due to these poorly fitted coefficients, for those languages, the regression with the lowest validation error is

regression III, which only includes character entropy and script type as predictors. The validation RMSE is 0.589, indicating that predicted byte premiums for languages with uncommon scripts are on average approximately 0.589 away from the true byte premiums. Given that byte premiums can range from below 0.75 to over 5.00, even this simple regression is a substantial improvement over a naive assumption that languages take equal bytes to encode information (i.e. byte premium 1.0).

2.1.6 Introducing the Tool

Finally, we introduce a Python tool that returns pre-computed or predicted byte premiums for any language pair. The tool is available at `https://github.com/catherinearnett/byte-premium-tool`. If both input languages are in our set of 1155 languages, the pairwise byte premium is computed from Equation 2.1 using our pre-computed byte premiums. Otherwise, the byte premium is computed from a user-provided parallel text (if available). If no parallel text is available, the tool asks for a small monolingual corpus in the novel language(s), from which it can compute the character entropy and bytes-per-character ratio per language, to use in the regressions from §2.1.4. Following the validation results in §2.1.5, the tool uses regression I, II, or III (in order of decreasing priority) for languages with common scripts. For languages with uncommon scripts, regression III is always used. Aside from character entropy (which is computed from the user-provided monolingual text), regression III requires only the script type for the novel language(s), which can easily be found on sites such as Wikipedia. Thus, our tool provides a simple interface from which to obtain the pairwise byte premium between any two languages, enabling

easy dataset size conversions.

2.1.7 Discussion and Conclusion

Measuring Dataset Sizes One implication of our work is that researchers currently may overestimate the amount of data that multilingual NLP models are trained on for non-Latin script languages (languages with high byte premiums). These languages are often already underrepresented in NLP (van Esch et al., 2022). For example, if it is reported that a model is trained on 1GB of Georgian data, then based on its byte premium of 4.34 relative to English, we should consider the model to be effectively trained on the Georgian equivalent of about 230MB of English data.

As a preliminary investigation into whether scaling training data quantities by byte premiums per language is indeed a "better" measure of training data quantity, we use this scaled measure to predict multilingual language model performance on various per-language benchmarks. We compile reported training data proportions (measured based on bytes) per language for existing massively multilingual models. We adjust each training data proportion by dividing the reported proportion by the byte premium for that language. After re-scaling to sum to 1.0, this provides the estimated effective proportion of data for each language. If adjusted data proportions are indeed "better" estimates of effective data quantities, then we expect them to predict downstream task performance better than the original reported training data proportions.

We evaluate ten models from three model families: XGLM (Lin et al., 2022), BLOOM (Scao et al., 2022), and mT0 (Muennighoff et al., 2023). We compile results

from XGLM 7.5B, four sizes of BLOOM (560M, 1.1B, 3B, 7.1B), and five sizes of mT0 (small, base, large, xl, xxl). We use benchmark scores from five multilingual benchmarks: XStoryCloze (Lin et al., 2022), XCOPA (Ponti et al., 2020), XNLI (Conneau et al., 2018), Wikipedia next word prediction (Guo et al., 2020), and XWinograd (Muennighoff et al., 2023). These benchmarks cover 22 languages: Arabic, Bulgarian, German, Greek, English, Estonian, French, Haitian Creole, Hindi, Indonesian, Italian, Japanese, Burmese, Portuguese, Russian, Spanish, Swahili, Telugu, Turkish, Urdu, Vietnamese, and Chinese (simplified and traditional). Benchmark scores are compiled from the Big Science evaluation results on Hugging Face.[9]

We fit two linear mixed effects models. Each predicts the benchmark score for each language (all scores between 0.0 and 1.0) from the training data proportion for that language (either the original proportion or those scaled according to our byte premiums) as well as language family, with random intercepts for model and task. We calculate the AICs of the two non-nested models, along with their relative log likelihoods (Wagenmakers and Farrell, 2004). While the the data proportions scaled by byte premiums better predict benchmark performance (lower AIC and higher log likelihood), it is not a significant difference ($p = 0.13$), using significance testing as in Wagenmakers and Farrell (2004).

Byte-Level Tokenization Our results also have implications for dataset tokenization. Previous work has argued that byte-level tokenizers enable more uniform treatment of different languages in a model (Zhang and Xu, 2022; Xue et al., 2022), but our byte premiums demonstrate that some languages may still be at a disadvantage

[9]https://huggingface.co/datasets/bigscience/evaluation-results

with byte-level tokenizers. Tokenization length inequalities can lead to higher costs, longer latencies, and restricted effective context lengths for some languages (Ahia et al., 2023; Petrov et al., 2023), in this case languages with high byte premiums.

Equitable Resource Costs Finally, languages with high byte premiums require more storage space than other languages to store comparable content, and they are likely to require higher bandwidth connections to transmit text content. In cases where storage is charged per (giga)byte or Internet connections are charged based on bandwidth and usage, uniform pricing rates across languages may lead to higher technology costs for low-resource language communities. While only a marginal step towards solving such issues, our work makes it possible to take byte premiums into account when measuring text data sizes across languages.

2.1.8 Acknowledgments

This chapter is a reprint of a paper published in the proceedings of the Special Interest Group on Under-resourced Languages (SIGUL) workshop at the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024).

Chapter 3

Morphological Alignment of Tokenization

The current chapter investigates the role of tokenization on downstream tasks that require representations of linguistic information, in this case subject-verb agreement. The specific aspect of tokenization I discuss is morphological alignment, that is whether the token boundaries align with morphological boundaries.

As a preface to this chapter, I introduce the ongoing discussion on optimal tokenizer design and evaluation, with an emphasis on morphological alignment. I discuss theories from linguistics and psycholinguistics, which I argue shed light on the desiderata of tokenizers for language models.

Tokenizers are generally designed for efficient segmentation, to be flexible enough to minimize the number of 'unknown' tokens, and to offer some compression. They are not designed to have linguistically interpretable or cognitively plausible

vocabularies. However, some researchers argue that morphologically aligned tokenization, i.e. tokenization that segments along morphological boundaries, is the gold standard for tokenization (Hofmann et al., 2022; Bauwens and Delobelle, 2024; Libovický and Helcl, 2024, *inter alia*). But there are reasons to believe this is not the case, independent from empirical evidence in NLP. I will first discuss evidence from linguistics, then evidence from psycholinguistics.

Linguistic Evidence Against Decomposition Hypothesis

Designing a tool to segment text into discrete morphemes using linear sequences is likely impossible. After decades of work focusing on English, even segmentation for English has not been solved. English, in fact, is far from the ideal language to apply this approach to be applied to. English is a fusional language, meaning that morphemes often encode more than one grammatical or semantic feature. Therefore, it is not trivial to segment words into discrete morphemes. Attempting to do so will lead to multiple possible segmentations, none of which will be obviously more correct than the others[1] (Lounsbury, 1953; Blevins, 2016).

For example, the regular nominal pluralization morpheme '-s' is often easily segmented (e.g. (1-a)), however, in the case of 'flies', it is ambiguous whether the best segmentation should be as in (1-b) or as in (1-c). While these forms are regular, the orthography introduces ambiguity for segmentation. In the case of irregular forms,

[1] "In a fusional language, if one seeks to arrive at constant segments... conflicts arise in the placing of the cuts. One comparison of forms suggests one placement, while another comparison suggests another. Often, in fact, no constant segment can be isolated at all which corresponds to a given constant meaning. Situations of this kind often permit of more than one solution according to different manners of selecting and grouping environments." (Lounsbury, 1953, p. 172 in Blevins, 2016)

e.g. ‘better’, there is no clear split between the lemma ‘good’ and comparative morpheme ‘-er’. This kind of approach to the segmentation of complex words, only words for languages with perfectly symmetrical morphological systems (Stevens and Plaut, 2022), of which there is no documented example.

(1) a. dog + s

b. flie + s

c. fli + es

(Nouri and Yangarber, 2016)

In order to fully implement this approach, one has to assume that "words can be fully decomposed for all languages" as if "[...] every language is ‘really’ agglutinative" (Hockett, 1987 in Blevins, 2016). Beyond English, non-concatenative morphology, periphrasis, and other morphological phenomena further challenge this approach.

Therefore, the decomposition hypothesis is practically difficult to implement and makes strong assumptions about the underlying structure of words, which does not completely reflect what is known about word formation crosslinguistically.

Psycholinguistic Evidence against morphological segmentation (decomposition hypothesis

Psycholinguistic evidence also challenges the usefulness of the decomposition hypothesis. Research on human visual language processing has grappled with some of the same issues which now related to tokenization. Subword tokenization was

developed in part because is too inefficient to store every possible token, and a language model must learn a token embedding for each token in its vocabulary. So each token takes up computationally expensive trainable parameters. For humans, the hypothesis that humans store every word form in their mental lexicon is called the full listing hypothesis (Butterworth, 1983; Bybee, 1995). Similar to tokenization, this is viewed to be inefficient for humans due to the memory demands.

The converse hypothesis is the parsing hypothesis or the decomposition hypothesis (Taft and Forster, 1976). Under this hypothesis, the mental lexicon stores lemmas, and complete word forms are computed during production. During comprehension, words are decomposed into their lemmas and morphemes.

Empirical psycholinguistics supports a hybrid model, the dual-route model (Baayen, 1995). Under this model, people store some high-frequency or non-compositional word forms whole in the mental lexicon, but lower-frequency items may be composed and decomposed from their morphemes during production and comprehension[2].

This dual-route hypothesis has been proposed as being relevant to tokenization (Hofmann et al., 2021), as many subword tokenizers store some high-frequency word forms whole as tokens, while less frequent forms are generally decomposed into multiple tokens. Accepting the dual-route hypothesis as a hypothesis for optimal tokenization challenges the strong decomposition approach to tokenization, as even when words are decomposable, it may be more efficient to store them as whole words. One reason for this is compression. It is twice as computationally expensive to process a word represented by two tokens as a word represented by a single token. The

[2]See Arcara et al. (2014) for an overview.

work in this chapter contributes to the literature on this issue.

3.1 Different Tokenization Schemes Lead to Comparable Performance in Spanish Number Agreement

Abstract

The relationship between language model tokenization and performance is an open area of research. Here, we investigate how different tokenization schemes impact number agreement in Spanish plurals. We find that morphologically-aligned tokenization performs similarly to other tokenization schemes, even when induced artificially for words that would not be tokenized that way during training. We then present exploratory analyses demonstrating that language model embeddings for different plural tokenizations have similar distributions along the embedding space axis that maximally distinguishes singular and plural nouns. Our results suggest that morphologically-aligned tokenization is a viable tokenization approach, and existing models already generalize some morphological patterns to new items. However, our results indicate that morphological tokenization is not strictly required for performance.

3.1.1 Introduction

In natural language processing (NLP) pipelines, **tokenizers** segment unstructured text into smaller, discrete constituents ("tokens") for further processing. Importantly, different tokenizers can incur performance and efficiency trade-offs. Assigning a unique token to each word in a corpus may lead to high-precision semantic representations, but the resulting models might be less robust to unseen words and require more computational resources.

Most existing tokenizers allow words to be decomposed into subword tokens (Sennrich et al., 2016; Kudo and Richardson, 2018). They can do so along morphological boundaries (e.g. *books* to ['book', '##s']), but this behavior is not guaranteed. Segmenting words into their lemmas and morphemes might simultaneously allow models to more robustly learn morphosyntactic patterns, more efficiently represent such patterns, and better generalize to novel words. (An analogous question concerning the storage of whole words vs. learning generalizable rules exists within human psycholinguistics research, e.g., Ullman, 2016).

In the current work, we ask whether and how the tokenization strategy employed facilitates successful language model predictions. We evaluate the effect of three types of plural noun tokenization in Spanish—single-token plurals, morphemically-tokenized plurals, and non-morphemically-tokenized plurals—in the context of a masked article prediction task (§3.1.4).[3] We focus on tokenization schemes for plural forms in Spanish, as it offers relatively simple and frequent ex-

[3]Note that this categorization scheme mirrors an approach taken in contemporaneous work, using the labels "vocab", "morph", and "alien", respectively.

amples of morphologically complex words. Spanish leverages two primary plural marking strategies, which are highly predictable for any given lemma. We specifically focus on cases where the plural form is composed of the singular form with the addition of '-s' or '-es'.

We find that tokenization schemes are differentially successful, although the effect is small, and article agreement accuracy is high across all tokenization types. Artificial tokenization schemes, where we coerce an initially single-token or non-morphemically-tokenized plural into a morphemic representation, leads to successful task performance, but does not improve performance beyond the original tokenization scheme. In an exploratory analysis, we compare singular and plural form embeddings across all tokenization schemes. We find axes with high overlap between all plural forms (regardless of tokenization scheme) and high discriminability between plural and singular forms, but other axes can still separate different plural tokenization schemes. This work contributes to a growing literature examining the impact of tokenization on the language modeling objective. Code and data are available: `https://github.com/catherinearnett/spanish-plural-agreement`.

3.1.2 Related Work

Several studies have investigated morpho-syntactic agreement in BERT-style models across multiple languages (Linzen et al., 2016; Mueller et al., 2020; Edmiston, 2020; Pérez-Mayos et al., 2021, inter alia), finding generally high agreement accuracy. In a subject-verb agreement task, however, BETO incurs a relatively high rate of agreement errors for certain Spanish nouns (despite the ability to extend number

agreement to novel words; Haley, 2020). It is unclear to what extent degraded performance is attributable to tokenization scheme, but the word "comanas"—listed as an example of a frequently mis-numbered word—is tokenized non-morphemically into [‘coman’, ‘##as’].

Indeed, recent work has demonstrated that morphologically-aware tokenization improves NLP model performance on a variety of downstream benchmarks (Park et al., 2020; Hofmann et al., 2021; Toraman et al., 2023; Jabbar, 2024; Uzan et al., 2024). Most relevantly, Batsuren et al. (2024) devise a tool to classify English words in terms of whether they are stored as single tokens ("vocab"), as multiple morphemic tokens ("morph"), or as multiple non-morphemic tokens ("alien"). The authors find that how multi-morphemic English words are tokenized is correlated with the language model's downstream performance on several tasks.

In line with Batsuren et al. (2024), our work investigates how the tokenization of Spanish nouns affects language model predictions involving a specific morphosyntactic rule, providing insight into how morphologically-aware tokenization affects NLP model performance.

3.1.3 Model and Data

All experiments use BETO, a Spanish pre-trained BERT model (Cañete et al., 2020) with 110M parameters trained on approximately 3B words. BETO uses a SentencePiece tokenizer (Kudo and Richardson, 2018) with a 32K vocab size.

Data

All plural nouns and their singular form lemmas were extracted from the AnCora Treebanks (Alonso and Zeman, 2016). Plurals were categorized according to their affix. Nouns ending in vowels use the plural suffix -s, while nouns ending in consonants use the suffix -es. Plurals were also annotated for their grammatical gender by a native Spanish speaker. Irregular nouns, misspellings, and words not listed in the Real Academia Española (RAE) online dictionary were excluded.

Identifying Tokenization Type

We created three lists of plurals: one-token (n=1247), multi-token morphemic (n=508), and multi-token non-morphemic (n=627). One-token plurals are stored as single tokens in the tokenizer's vocabulary. We then categorized multi-token plurals as morphemic or non-morphemic. If tokenization followed morpheme boundaries (e.g., *naranjas* as ['naranja', '##s']), the noun was categorized as morphemic; if not, it was categorized as non-morphemic (e.g., *neuronas* is tokenized as ['neuro', '##nas']).

Relationship of Tokenization to Frequency

Using oral frequency measures for 2071 target plural wordforms available in a corpus of over 3M spoken words (Alonso et al., 2011), we examined the relationship between a wordform's frequency and how it was tokenized. A linear model predicting Log Frequency from Tokenization Scheme explained significant variance [$R^2 = 0.33$]. With MORPHEMIC level as a reference class (i.e., intercept), the NON-MORPHEMIC

plural nouns were significantly less frequent [$\beta = -0.18, SE = 0.03, p < .001$], while the SINGLE-TOKEN plural nouns were significantly more frequent [$\beta = 0.59, SE = 0.03, p < .001$]. As expected, the frequency of a wordform was likely a major factor in how it was tokenized.

Due to the relationship between tokenization scheme and wordform frequency, we carried out several supplementary analyses to determine the extent to which frequency was a confound in the results presented in Section 3.1.4. We found two key results: first, BETO's predictions were indeed more accurate for more frequent wordforms; second, however, BETO's predictions were still more accurate for some of the original tokenization schemes than others, even controlling for wordform frequency.

We ran a follow-up analysis asking whether the Log Frequency of a wordform was predictive of agreement success. This analysis had two key goals. First, because Log Frequency was correlated with Tokenization Scheme, we aimed to determine whether the effect of Tokenization Scheme on agreement success was in fact due to effects of token frequency. Second, we were independently interested in whether the language model made better predictions for more frequent wordforms.

We fitted a linear mixed-effects model including fixed effects of Tokenization Scheme, Word Number, and Log Frequency, as well as interactions between Word Number and Tokenization Scheme and between Word Number and Log Frequency. We also included random intercepts for word lemma and sentence. This model explained significantly more variance than a model omitting only the interaction between Log Frequency and Word Number [$\chi^2(1) = 17.89, p < .001$]. The interaction was negative [$\beta = -0.35, SE = 0.08, p < .001$], i.e., the plural article log-odds were

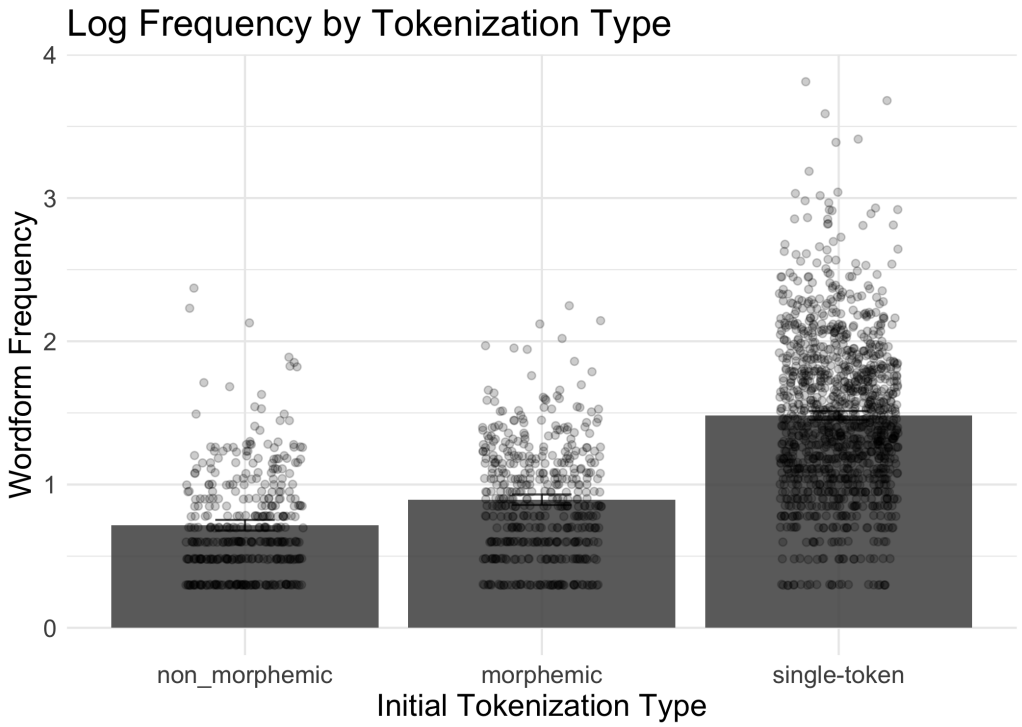

Figure 3.1: Single-token plurals were significantly more frequent than those tokenized according to morphemic boundaries, which were more frequent than those tokenized according to non-morphemic substrings.

more negative for more frequent singular nouns. In other words, the language model made better predictions for more frequent nouns than less frequent nouns.

The full model also explained more variance than a model omitting the interaction between Word Number and Tokenization Scheme [$\chi^2(2) = 11.24, p = .004$]. This indicates that even controlling for wordform frequency, there was an independent effect of how the wordform was initially tokenized on the success of the language model's article predictions.

Artificial Tokenization Procedure

To investigate the effect of tokenizing a wordform at the morpheme boundary, we artificially tokenized single-token and multi-token non-morphemic plural nouns by concatenating the token for the appropriate affix (e.g., "##es") onto the token(s) for the singular noun (Table 3.1).

Table 3.1: Artificial tokenizations for the words *mujeres* 'women' (*mujer*), and *patronos* 'employers' (*patrono*).

Morpheme Boundary	**Original Tokenization**	**Artificial Tokenization**
mujer+es	['mujeres']	['mujer', '##es']
patrono+s	['patr, '##onos']	['patr, '##ono', '##s']

3.1.4 Study: Article-Noun Agreement

Our primary research question concerned the impact of the original tokenization (Tokenization Scheme) on an article agreement task, similar to that implemented by Linzen et al. (2016). In Spanish, articles must agree with the *number* of the noun (e.g., *la mujer* vs. *las mujeres*); learned representations for the target noun should thus be conducive to predicting article number. We asked:

1. How does the initial tokenization scheme of a plural noun impact the language model's ability to predict the correct article?

2. Does our *artificial* tokenization scheme provide sufficient information to facilitate successful agreement?

3. How does the success of our artificial tokenization scheme compare to the original tokenization scheme for those nouns?

Method

Agreement was assessed by taking the logarithm of the relative probability of a plural vs. singular article as predicted by a given noun. For a given wordform (e.g., *mujeres*), a positive log-odds indicated a higher probability was assigned to the plural article, while a negative log-odds indicated a higher probability was assigned to the singular article. A *singular* noun should be associated with a more negative log-odds, while a *plural* noun should be associated with a more positive log-odds. We considered both DEFINITE and INDEFINITE articles (ARTICLE TYPE) for each wordform; the log-odds calculation was performed separately for each type.

Within the sequence of model inputs, only the article token was masked, and special tokens ([CLS], [SEP]) were included, as in the examples below:

- Example model inputs for original single-tokenizations: "[CLS] [MASK] mujeres [SEP]"
- Example model inputs for artificial (morphemic) tokenizations: "[CLS] [MASK] mujer ##es [SEP]"
- Example model inputs for original non-morphemic multi-tokenizations: "[CLS] [MASK] patr ##onos [SEP]"
- Example model inputs for artificial (morphemic) tokenizations: "[CLS] [MASK] patr ##ono ##s [SEP]"

For each sequence of inputs independently, we obtain BETO's output logits over the target token corresponding to the (1) definite singular, (2) indefinite singular, (3) definite plural, and (4) indefinite plural articles. We subsequently apply softmax normalization to each token's logits to obtain the log probabilities of filling the masked item with a particular article.

Accounting for the different presentations of each wordform (i.e., definite vs. indefinite article; original vs. artificial tokenization), our final dataset had 13,276 observations in total, each with an accompanying *log-odds* ratio. All data and visualizations were analyzed in R; mixed effects models were fit using the *lme4* package (Douglas Bates et al., 2015). Maximal random effects structures were fit where possible, and reduced as needed for model convergence.

Results

Impact of Initial Tokenization We first asked whether the original tokenization scheme used for plural nouns affected successful agreement. We fit a mixed model with Log Odds as a dependent variable, fixed effects of Tokenization Scheme and Word Number (and an interaction between the two), fixed effects of Article Type, and random intercepts for each word lemma and sentence. This model explained significantly more variance than a model omitting only the interaction $[\chi^2(2) = 6.54, p = .04]$, suggesting that different tokenization schemes were differentially successful in predicting the appropriate article.

However, as depicted in Figure 3.2, this effect was quite small. Accuracy was near ceiling for all tokenization types, i.e., the Log Odds was larger than 0 for plural

nouns and smaller than 0 for singular nouns (see also Table 3.2). Thus, our results do not suggest that morphologically-aligned tokenization is required for good agreement performance.

Table 3.2: Accuracy scores for *plural nouns* only, using either the original tokenization scheme for that class of nouns or the artificially-induced morphemic scheme.

Original Tokenization	**Original**	**Artificial**
Morphemic	0.97	—
Non-morphemic	0.98	0.96
Single-Token	0.98	0.97

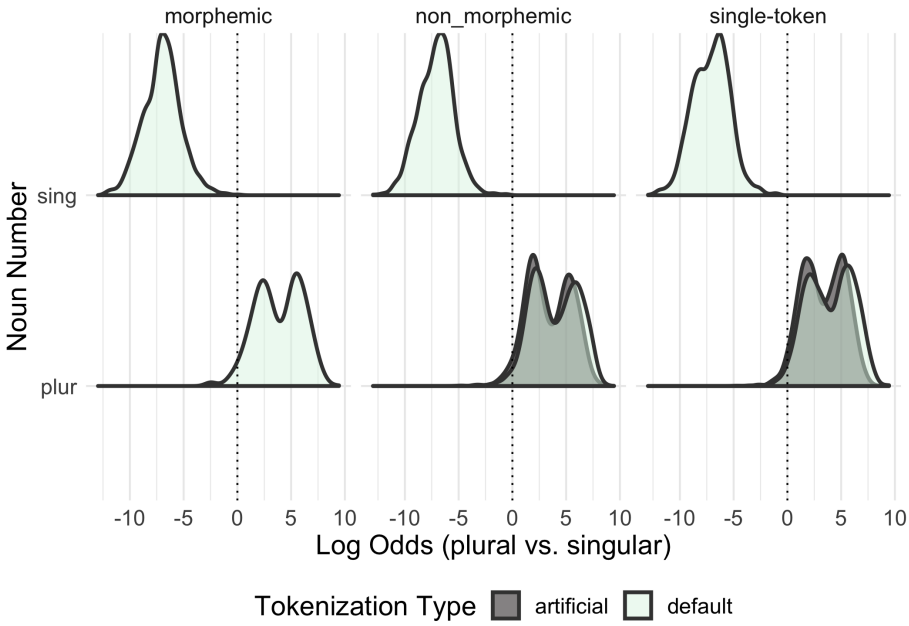

Figure 3.2: Log-odds varied significantly as a function of noun number (*singular* vs. *plural*). The extent of this variance interacted (weakly) with initial tokenization (*morphemic* vs. *non-morphemic* vs. *single-token*) and with whether the *original* or *artificial* tokenization procedure was used. Larger log-odds indicate higher probabilities of the plural article.

Success of Artificial Tokenization Next, we artificially tokenized plural nouns that would otherwise be tokenized non-morphemically or as a single-token. To quantify the success of this procedure, we fitted a linear mixed-effects model predicting Log Odds with fixed effects of Article Type, Word Number, Tokenization Scheme, and Affix ("##s" or "##es"), as well as random intercepts for word lemma and sentence.

This model explained significantly more variance than a model omitting only Word Number [$\chi^2(1) = 11988, p < .001$], indicating that the artificial tokenization procedure still led to good article number agreement performance: Log Odds were significantly different for singular nouns and artificially-tokenized plural nouns (see also Figure 3.2 and Table 3.2).

Comparing Default vs. Artificial Tokenization Schemes Finally, restricting our analysis to plural forms, we asked whether a higher Log Odds was assigned to *artificially tokenized* plural nouns than ones using the default scheme. We fitted a linear mixed-effects model with fixed effects of Tokenization Scheme (artificial or original), Affix, and Original Tokenization Scheme (as well as random intercepts for word lemma, sentence, and wordform, and by-lemma random slopes for Tokenization Scheme). This model did explain more variance than a model omitting only Tokenization Scheme [$\chi^2(1) = 141.81, p < .001$]. Critically, however, the Log Odds for the artificially tokenized plural nouns was *lower* ($M = 3.38, SD = 2$) than when using the default tokenization ($M = 3.95, SD = 2.15$). In other words, the artificially-induced morphemic tokenization was successful, but less so than relying on the original scheme for those nouns.

3.1.5 Linear Discriminant Analysis (LDA)

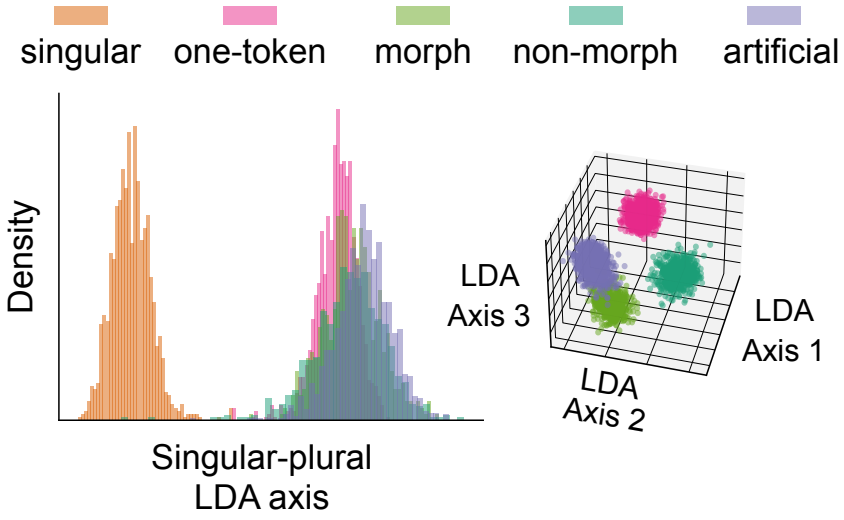

Figure 3.3: LDA for singular and plural embeddings reveals axes of overlap (*left*) and discriminability (*right*) for differentially tokenized plural forms.

To identify potential causes for the observed agreement patterns across noun types (singular vs. different plural tokenizations), we considered the embeddings of those nouns in the language model representation space. We took each noun's mean embedding across the last four (out of twelve) BETO Transformer layers, averaging over all tokens in the noun. To minimize confounds from averaging embeddings over different numbers of tokens, we considered only two-token plurals in all multi-token scenarios for embedding analyses.

We first identified the linear axis that maximally separated single-token singular from plural nouns. To do this, we ran linear discriminant analysis (LDA) with two classes of embeddings: singular nouns (all single-token) and single-token plural

nouns.[4] We then projected all noun representations linearly onto this axis, essentially projecting each embedding into a single value. As expected, we found that singular nouns clustered separately from plural nouns (Figure 3.3, *left*). Notably, all types of plurals (single-token, artificially tokenized, two-token morphemic, and two-token non-morphemic) patterned together and were not linearly discriminable along this axis. This suggests that the model could rely on similar number agreement mechanisms for different types of plurals, but future work would need to demonstrate causal impacts of this singular-plural axis on number agreement predictions (e.g. as in Mueller et al., 2022b).

While the singular-plural LDA axis mapped different plural types to similar values, other axes could separate embeddings for the different plural types. We used LDA to identify the three linear axes that maximally separated the four types of plurals. As shown in Figure 3.3 (*right*), single-token plurals and two-token non-morphemic plurals were separable from one another and from all other plural types. The artificial and default morphemic plurals had distinct clusters, but they were not entirely separable from one another. This indicates that even though the artificial tokenization was never seen by the model during training, the representations were still quite similar (e.g. due to the presence of the '##s' or '##es' token). The slight separation between these clusters may be driven either by frequency effects or by veridical differences in how the models represent number in the two plural types.

[4]Given n sets of representations, LDA computes $n-1$ directions in the language model representation space that maximize separation between the sets.

3.1.6 Discussion and Conclusion

We assessed whether distinct tokenization schemes impacted the ability of BETO (a Spanish language model) to predict appropriate articles for Spanish plural nouns. Single-token representations facilitated slightly better predictions overall. However, the model did show evidence of generalization consistent with having learned morpheme-like "rules": artificially re-tokenizing plural nouns along morpheme boundaries produced representations amenable to article prediction—despite the language model never having previously observed that sequence of tokens (see Figure 3.2)—though this approach was slightly less accurate than relying on the original tokenization scheme. This provides further insight into work on language models generalizing morphological patterns (Haley, 2020); however, this does not work equally well for all languages or models (Weissweiler et al., 2023a).

Notably, the similar agreement performance across single-token, morphological, non-morphological, and artificially-tokenized plurals could indicate multiple different agreement mechanisms in the model. At least on this task, tokenization along morpheme boundaries was not correlated with improved agreement performance; this is in contrast to other work suggesting that morphologically aware tokenization improves performance, e.g., in machine translation (Macháček et al., 2018) or similarity judgments (Batsuren et al., 2024). Future work might apply causal interventions on different embedding axes (as found in §3.1.5), to determine the extent to which the same model subnetworks are involved in number agreement for different types of plural tokenizations, shedding light on the impacts of tokenization on language model processing.

3.1.7 Limitations

A key limitation of the current work is scope. Future work could consider additional morphological phenomena, additional languages, and a larger range of language models or tokenization schemes. A second limitation is that the language model's performance was near-ceiling for each category considered. It is possible that different tokenization strategies do in fact impact agreement performance under more challenging conditions, but that the near-ceiling performance on this task made it difficult to detect those differences. Future work could work to develop more challenging tasks for which the model is not at ceiling (as in Linzen et al., 2016), or for which variance in how multi-morphemic words are parsed might be expected to contribute more to downstream performance (Batsuren et al., 2024). Finally, our work does not demonstrate the extent to which different tokenizations rely on the same internal mechanisms for agreement in the model (§3.1.6), which is a valuable direction for future work.

3.1.8 Acknowledgments

This chapter is a reprint of a paper published in the proceedings of the Special Interest Group on Computational Morphology and Phonology (SIGMORPHON) at 2024 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2024).

Chapter 4

Do Crosslinguistic Typological Differences Drive Inequities in NLP Performance?

There is a pervasive assumption that language technologies developed for English can be applied to other languages, as long as there is sufficient pre-training data (Bender, 2011). In this chapter, I ask whether there is empirical evidence to contradict this assumption. There are observed differences in performance (as discussed in Section 1.2). Language model performance varies greatly between languages, often such that language model performance for languages more dissimilar to English is worse than that for languages more similar to English. Therefore, it has been proposed that there are structural properties of a language that might lead to better or worse language model performance. But is this because of inherent language dif-

ferences or is it simply due to a lack of data? In this chapter, I investigate whether there are particular features of a language that are correlated with lower performance, specifically morphological typology.

This questions is of broad interest for crosslingual and multilingual NLP, as it is important to know whether there are deep, architectural design choices in NLP that systematically benefit English over other languages. The results from this chapter bear on the discussion of morphologically aligned tokenization, which was discussed in Chapter 3.

4.1 Why do language models perform worse for morphologically complex languages?

4.1.1 Abstract

Language models perform differently across languages. It has been previously suggested that morphological typology may explain some of this variability (Cotterell et al., 2018). We replicate previous analyses and find additional new evidence for a performance gap between agglutinative and fusional languages, where fusional languages, such as English, tend to have better language modeling performance than morphologically more complex languages like Turkish. We then propose and test three possible causes for this performance gap: morphological alignment of tokenizers, tokenization quality, and disparities in dataset sizes and measurement. To test the morphological alignment hypothesis, we present MorphScore, a tokenizer eval-

uation metric, and supporting datasets for 22 languages. We find some evidence that tokenization quality explains the performance gap, but none for the role of morphological alignment. Instead we find that the performance gap is most reduced when training datasets are of equivalent size across language types, but only when scaled according to the so-called ‘’byte-premium"—the different encoding efficiencies of different languages and orthographies. These results suggest that languages of particular morphological types are not intrinsically advantaged or disadvantaged in language modeling. Differences in performance can be attributed to disparities in dataset size. These findings bear on ongoing efforts to improve performance for low-performing and under-resourced languages.

4.1.2 Introduction

An enduring goal in NLP is to develop language-general systems that achieve equal performance on all languages (Bender, 2011). Yet to date performance on languages other than English and a small number of high-resource languages remains extremely poor (Joshi et al., 2020; Ranathunga and de Silva, 2022; Søgaard, 2022; Atari et al., 2023; Ramesh et al., 2023). This has been attributed to a lack of research on non-English languages (Blasi et al., 2022), a lack of training data, and the possibility that evaluations are skewed towards high-resource languages (Choudhury, 2023).

Beyond these systemic biases, it's also possible that certain linguistic features lead to higher or lower language modeling performance. Specifically, it has been

All code and data for this paper available below.
`https://osf.io/jukzd/?view_only=3d0d491d24074215a0ab81f72a693c16`

proposed that languages with more complex morphology are harder to model (Cotterell et al., 2018; Park et al., 2021b). Languages with more inflectional classes are morphologically more complex, and thus harder to predict. This can be described in terms of enumerative complexity (Ackerman and Malouf, 2013).

Greater morphological complexity may lead to worse language model performance, as morphologically rich languages tend to have a large number of very infrequent word forms produced by combinations of morphemes, which leads to data sparsity (Shin and You, 2009; Bender, 2011; Botev et al., 2022). This claim finds empirical support in Gerz et al. (2018a), who demonstrated over a sample of 50 languages that morphologically rich (agglutinative) languages performed worse than less morphologically rich (fusional) languages. In the current work (§4.1.4), we replicate this analysis and extend it to much larger transformer models, both in monolingual and multilingual settings. We, too, find a robust performance gap between agglutinative and fusional languages.

This effect is surprising, as there are reasons to think that agglutinating languages should be *easier* for language models to learn. In studies on first language acquisition, children are observed to acquire more complex morphological systems earlier, especially systems that are uniform and transparent (Dressler, 2010). This may be due to the fact that the form-meaning correspondences in these systems are more transparent, and thus more informative (Slobin, 1973, 2013, 2001; Dressler, 2010). By adulthood, there are no observed cross-linguistic differences in the level of acquisition of different languages according to morphological typology. Therefore, there is no linguistic evidence that would predict that any language should be harder

to learn than any other language.

Identifying the causes for this performance gap could permit improved performance for morphologically rich languages (which are often low-resource) and reduce the performance inequity, potentially enabling users and researchers to be better able to use and do research on language models (Khanuja et al., 2023) in their own languages. We evaluate three possible explanations.

Hypothesis 1: Tokenization is not Morphologically Aligned

When the token boundaries for a given word line up with its morpheme boundaries, that tokenization is morphologically aligned. For example, the word 'books' in English is composed of the root 'book' and the plural morpheme '-s'. A morphologically aligned tokenization would be ['book', 's']. By contrast, ['boo', 'ks'] and ['b', ooks'] would be morphologically misaligned tokenizations.

Morphological alignment of the tokenizer – or lack thereof – could impact language modeling performance, especially for morphologically rich languages. For these languages, relatively frequent morphemes are combined to create a large number of unique word forms, which may be rare or completely novel. If the tokenizer does not segment words along morphological boundaries, it may be difficult for the language model to efficiently learn and represent the structure of the language. Additionally, this may be further exaggerated for morphologically complex languages, which tend to have longer words.

This hypothesis would predict that agglutinative languages have less morphologically aligned tokenizers than fusional languages and that morphological alignment

negatively correlates with metrics of language model performance. To test this hypothesis, Section 4.1.5 introduces **MorphScore**, tokenizer evaluation for morphological alignment in 22 languages. To our knowledge, this is the first such multilingual evaluation for morphological alignment of tokenizers.

Hypothesis 2: Tokenization is Worse

Second, morphologically rich languages might tend to engender lower quality tokenizations. There is no current consensus on how to evaluate intrinsic tokenization quality (Zouhar et al., 2023; Chizhov et al., 2024). But compression is one of the most widely used metrics (Gallé, 2019; Rust et al., 2021, *inter alia*). It is usually measured as sequence length – the number of tokens needed to encode a sequence – or corpus token count (CTC; Schmidt et al., 2024). Better compression has been linked to better language modeling performance because it allows for more language data to fit into a fixed sequence length (Gallé, 2019; Liang et al., 2023; Dagan et al., 2024; Goldman et al., 2024); however, there is some evidence to suggest that compression is not directly linked to performance (Deletang et al., 2024; Schmidt et al., 2024).

Agglutinative languages might have worse compression on average because words tend to be longer (Fenk-Oczlon and Fenk, 1999; Berg et al., 2022) and there are more unique word forms (Sandra, 1994). In Turkish, for example, a single root may have millions of unique word forms (Hakkani-Tür et al., 2002). It is therefore less likely that the tokenizer will store whole words in its vocabulary, instead representing words using multiple tokens. This in turn may lead to worse compression and thus worse performance. If we find worse compression for agglutinative languages than

for fusional languages, this may indicate that suboptimal compression is related to the performance gap.

Another proposed metric of tokenization quality is Rényi entropy (Zouhar et al., 2023), which measures how evenly distributed token frequencies are over the whole vocabulary, penalizing very high- and very low-frequency tokens. Rényi entropy has been shown to be predictive of downstream task performance (ibid). Because of their larger number of low-frequency word forms, it is possible that agglutinative languages have higher numbers of low-frequency tokens (specific inflectional forms) or higher numbers of high-frequency tokens (very high frequency morphemes used in many different word forms) than fusional languages. Therefore if agglutinative languages display worse (higher) Rényi entropy than fusional languages, this could indicate that inefficient token frequency distribution contributes to the performance gap.

In Section 4.1.6, we collect both compression and Rényi entropy and test whether agglutinating languages have worse compression and Rényi entropy, which would suggest that aspects of tokenization quality are driving the performance gap.

Hypothesis 3: Less Training Data

The role of data quantity for pre-trained language models is uncontroversial: the more, the better. In some cases, increasing data can improve performance more than increasing model size (Hoffmann et al., 2022). Western European high-resource languages tend to be less morphologically rich, and correspondingly, many morphologically rich languages are low-resource. Morphologically rich languages have less

annotated data (Botev et al., 2022) and are less well researched. According to a survey by Blasi et al., despite having more speakers than most European languages, morphologically complex languages like Bengali, Swahili, and Korean have only a small number of studies. German, Romanian, French, and Italian have been better studied, despite having many fewer speakers (Blasi et al., 2022, Table 2). Therefore, data scarcity may be driving the observed performance gap between agglutinative languages and fusional languages.

Furthermore, recent work has shown that there are disparities in the number of bytes needed to convey the same amount of information in different languages (*byte premium*; Arnett et al., 2024a), due to orthographic encoding and linguistic reasons. Morphologically rich languages are more often written with non-Latin scripts, which require more bytes to be represented in common encoding standards like UTF-8. Morphologically rich languages also have longer words, which may amplify the effect. Byte premiums may thus exacerbate the data scarcity problem, and agglutinative languages may be trained on effectively less data even than it currently seems. Section 4.1.7 asks whether monolingual language models trained on byte-premium-scaled text demonstrate the previously observed performance gap.

4.1.3 Background

Morphological Typology

The field of morphological typology seeks to categorize languages according to their word formation strategies (Brown, 2010). Some languages primarily use words composed of a single morpheme or a small number of morphemes. Other lan-

guages incorporate many morphemes into a single word. This paper focuses on two types of languages: fusional and agglutinative languages. Fusional languages tend to encode multiple morpho-syntactic features into a single morpheme, where agglutinative languages tend to use different morphemes to represent each feature (Plank, 1999; Haspelmath, 2009; Dressler, 2010). As a result, agglutinative languages also tend to be polysynthetic (having words composed of many individual morphemes; Baker, 1996). For example, Turkish has separate plural and accusative morphemes, but in English, the root, tense, number, and person may all be loaded onto a single morpheme (Exs. (1) and (2)).

(1) tarla-lar-ı (Turkish)
field-PL-ACC
(Plank, 1999)

(2) are
be-PRES.2PL (English)

Typological categorization is much more complex than this binary categorical distinction. In order to connect this work with previous studies, it is helpful to use a very coarse view of morphological type; however, these properties are gradient. Languages may have both fusional and agglutinative properties (and properties of other morphological types, too). See Plank (1991; pp. 11-16) for discussion on this point.

Morphologically Aligned Tokenization

There is an area of active research on the relationship between morphological alignment of tokenizers and how it relates to language model performance. Work in this area often stems from the assumption that morphologically aligned tokenization is the gold standard for tokenization (Hofmann et al., 2022; Bauwens and Delobelle, 2024; Libovickỳ and Helcl, 2024, *inter alia*). Morphologically-aware or aligned tokenization has been argued to lead to more meaningful tokens, which in turn leads to better performance (Banerjee and Bhattacharyya, 2018; Klein and Tsarfaty, 2020; Tan et al., 2020; Hofmann et al., 2021, 2022; Minixhofer et al., 2023; Bauwens and Delobelle, 2024). There is empirical evidence from several languages to support this claim, e.g. English (Jabbar, 2024), Korean (Lee et al., 2024) , Latvian (Pinnis et al., 2017), Arabic (Tawfik et al., 2019), Japanese (Bostrom and Durrett, 2020), Hebrew (Gueta et al., 2023), Kinyarwanda (Nzeyimana and Niyongabo Rubungo, 2022), and Uyghur (Abulimiti and Schultz, 2020). However, these efforts are limited by the availability of morphologically annotated datasets (Minixhofer et al., 2023), which are often only available for a small number of relatively high-resource languages.

Some evidence also exists to the contrary (Zhou, 2018; Minixhofer et al., 2023; Gutierrez-Vasques et al., 2023), even for some of the same languages. Work on German and Czech (Macháček et al., 2018), Nepali, Sinhala, and Kazakh (Saleva and Lignos, 2021), Korean (Choo and Kim, 2023), Turkish (Kaya and Tantuğ, 2024), and Spanish (Arnett et al., 2024b) did not show any benefit of morphologically aligned tokenization. This is consistent with other work, e.g. Uzan et al. (2024), showing that BPE, which generally performs best on metrics such as compression, has the

least morphologically meaningful tokens compared to other tokenization algorithms.

4.1.4 Evidence for a Performance Gap

This section describes three analyses that show lower performance for agglutinative languages. Previous analyses, which demonstrated evidence for the performance gap between fusional and agglutinative languages, all had significant confounds. We extend previous work by additionally controlling for amount of training data and extending to models which–as they are much larger and use the transformer architecture–better represent the state of the field.

Reanalysis of Gerz et al. (2018a)

Gerz et al. (2018b) analyzed a multilingual LSTM trained on 50 languages and found that fusional languages categorically outperformed agglutinative languages. This seminal finding is nevertheless limited in ways. It did not control for the number of training tokens, which was different for each language. We addressed this in a replication of the analysis on the original data, fitting a full linear model in R with morphological type and number of training tokens as fixed effects, predicting perplexity. We fit a reduced model with only number of training tokens as a fixed effect. An ANOVA showed that the full model explained more variance in the data than the reduced model ($F(3, 45)=5.221$, $p=0.004$)). After controlling for number of training tokens, there is still a significant effect of morphological type, where agglutinative languages had higher perplexities than fusional languages.

Multilingual Models

Evidence from Gerz et al. (2018a) comes from just one model. To extend this work, we test a number of more contemporary multilingual language models, including XGLM (Lin et al., 2022), BLOOM (Scao et al., 2022), mT0 (Muennighoff et al., 2023), MaLA (Lin et al., 2024), and LLaMA2 (Touvron et al., 2023b).

We test these models across a variety of benchmarks: commonsense reasoning benchmark scores from XStoryCloze (Lin et al., 2022), XCOPA (Ponti et al., 2020), XNLI (Conneau et al., 2018), Wikipedia (Guo et al., 2020), and XWinograd (Muennighoff et al., 2023) reported in the BigScience BLOOM evaluation results[1] and the SIB-200 benchmark (Adelani et al., 2024), as reported in the release paper.

We combine all of the benchmark scores into one dataset. All scores are on a scale between 0 and 1. We use language family information from the WALS database (Dryer and Haspelmath, 2013) and annotate the morphological type according to grammars and linguistic articles about each language. For each language model, we calculate the proportion of training data for each language according to reported data quantities in tokens or bytes. If languages were upsampled for model training, we include upsampled proportions.

We fit a full linear mixed effects model in R (Bates, 2010) predicting benchmark score with morphological type and language family as fixed effects and model and task as random effects. We fit a reduced model that is the same as the full model, except without morphological type as a predictor. We run an ANOVA to compare model fit. We find that the full model (with morphological type as a fixed effect)

[1] `https://huggingface.co/datasets/bigscience/evaluation-results`

explains more variance than the reduced model ($\chi^2(3) = 149.16$, $p<0.001$). Even after controlling for amount of training data, language family, model, and benchmark task, there is still a significant effect of morphological type, where fusional languages show better performance than agglutinative languages.

Monolingual Models

Both of the previous analyses measure performance of multilingual models. None of these models had controlled or balanced amounts of training data for the languages they were trained on. This introduces a confound, because European languages are typically both higher resource and fusional. The lower-resource languages in this sample were more likely to be agglutinative. In this final analysis of the performance gap, we compare performance of a suite of 1,989 monolingual models from Chang et al. (2023a), covering 252 languages, which were trained on matching numbers of tokens. For each language, there are up to 12 models, with up to three different model sizes and four different training corpus sizes. The three model sizes were tiny (4.6M parameters), mini (11.6M parameters), and small (29.5M parameters). The four dataset sizes were low-resource (1M tokens), medlow-resource (10M tokens), medhigh-resource (100M tokens), and high-resource (1B tokens). Perplexities were calculated using 500k held-out tokens. We use the same language family and morphological type data as in §4.1.4.

We use perplexity as a metric of performance. This is the only existing evaluation metric for all the languages represented by these models.

We fit a full linear regression with morphological type, model size, and dataset

size as predictors. We also fit a reduced model with only model size and dataset size as predictors. We use an ANOVA to compare the fit of these two models. We find that morphological type explains variance above and beyond the other two predictors ($\chi^2(3) = 28.809$, $p<0.001$). We also fit full and reduced models with language family as an additional predictor. Even after accounting for language family, morphological type still explains additional variance ($\chi^2(3) = 3.3324$, $p=0.02$).

Morphological type is predictive of performance after controlling for model size and data amounts, which supports the other analyses.

Interim Discussion

Using both perplexities and benchmark scores as evaluation metrics, and evaluations from monolingual and multilingual models, we found a robust performance gap between agglutinative languages and fusional languages. This evidence amplifies prior work by using more evaluation metrics for more languages, with more contemporary multilingual and monolingual models trained with balanced training data.

The following sections test three factors that may be driving this gap, corresponding to the three hypotheses above: morphological alignment of the tokenizers (§4.1.5), tokenization quality (§4.1.6), and disparities in data measurement (§4.1.7).

4.1.5 H1: Morphological Alignment

Does differential morphological alignment of tokenizers in languages with more or less complex morphology explain their performance gap? We present a new evaluation framework, called MorphScore, which permits a comparison of mor-

phological alignment across tokenizers and languages. We evaluate monolingual tokenizers for 22 languages and analyze the relationship between MorphScore and morphological type. Code and datasets for MorphScore are available on GitHub: `https://github.com/catherinearnett/morphscore`.

MorphScore: Evaluating Morphological Alignment of Tokenizers

Calculating MorphScore. To evaluate a tokenizer's MorphScore for each word in a test set, we assign a value of `1` if the tokenizer places a token boundary at the morpheme boundary of interest, regardless of other token boundaries. We assign a value of `0` if there is not a token boundary at the morpheme boundary of interest. We exclude items which contain no token boundaries (i.e. the entire word form is in the tokenizer's vocabulary), so as not to penalize the tokenizer for not segmenting the word. MorphScore is the mean of the assigned values across the dataset for a given language. See Table 4.1 for examples.

Languages. MorphScore uses datasets of morphologically annotated words. We created datasets for 22 languages, which are listed in Appendix B.1. Half are agglutinative languages and half are fusional, according to grammars and descriptions of the languages. Language selection was also balanced for resource level, where about half of the languages of each morphological type are higher-resource, and half lower-resource. The sample was designed to be as diverse as possible in terms of language family and writing system, given the other constraints. Note that all fusional languages in the sample are Indo-European, which reflects the distribution of fusional languages in the world's languages, but not all Indo-European languages are

fusional (e.g. Armenian). Among the Indo-European languages, two are from the Indic branch (Gujarati, Urdu) and a variety of subgroups are represented: Slavic (Bulgarian, Slovenian, Croatian), Baltic (Lithuanian), Hellenic (Greek), Armenian, Germanic (Swiss German, Icelandic), and Celtic (Irish). The other language families represented in the sample are Japonic, Koreanic, Dravidian, Kartvelian, Austronesian, Turkic, Niger-Congo, and Uralic, as well as an isolate (Basque).

Datasets. Each dataset is composed of words with their morpheme boundary annotations from Universal Dependencies[2] (UD) or UniMorph[3]. Words in MorphScore do not contain any umlaut or suppletion and the whole word form can be composed of the lemma and the morpheme (or the two morphemic units annotated). Most of the datasets only had one morpheme boundary annotation per word, with the exception of the Korean datasets. For Korean, when multiple morpheme boundaries were annotated, we chose the left-most boundary. We deduplicated items, and chose a random sample of 2000 for sets where there were more than 2000 items. We only included languages with at least 100 items.

Tokenizers

We use the monolingual tokenizers from Chang et al. (2023a), which are from the same models used for perplexities in §4.1.4. Each tokenizer is a SentencePiece (Kudo and Richardson, 2018) tokenizer with a vocabulary size of up to 32k. Each tokenizer is trained on 10k lines of text randomly sampled from the model training

[2] `https://universaldependencies.org/`
[3] `https://unimorph.github.io/`

Table 4.1: Example items with morphemic segmentations and tokenizations with MorphScores according to their morphological alignment.

Language	Word	Source	Segmentation	Score
Basque	aldiz	morphemic	aldi + z	
		Tokenizer 1	['al', 'diz']	0
		Tokenizer 2	['aldi', 'z']	1
Croatian	suučesnika	morphemic	suučesnik + a	
		Tokenizer 1	['su', 'uče', 's', 'nika']	0
		Tokenizer 2	['su', 'u', 'če', 's', 'nika']	0
Icelandic	samráðs	morphemic	samráð + s	
		Tokenizer 1	['samráð', 's']	1
		Tokenizer 2	['samráðs']	exclude
Greek	Αδριανής	morphemic	Αδριανή + ς	
		Tokenizer 1	['A', 'δ', 'ριανής']	0
		Tokenizer 2	['A', 'δρ', 'ιανή', 'ς']	1

data.

Results

We evaluate the tokenizers on their corresponding MorphScore dataset. MorphScores are reported in full in Table 4.2. In order to address Hypothesis 1, we first conduct a two sample t-test to evaluate whether agglutinative languages have lower MorphScores than fusional languages. This would be consistent with the explanation that tokenizers are more likely to fail to align token boundaries with morpheme boundaries in agglutinative languages. To the contrary, we find that agglutinative languages have higher MorphScores (M=66.3%) than fusional languages (M=53.3%), a significant difference (t(20.874)=2.950, p=.008).

We also tested for a negative correlation between MorphScore and perplexity, such that better MorphScores were correlated with better performance. We fit a

Table 4.2: MorphScore results from Section 4.1.5.

Lang	**Lang. Name**	**MorphScore**	**Morph. Type**
hye_armn	Armenian	0.634	agg
eus_latn	Basque	0.724	agg
bul_cyrl	Bulgarian	0.584	fus
ceb_latn	Cebuano	0.806	agg
eng_latn	English	0.781	fus
kat_geor	Georgian	0.660	agg
ell_grek	Greek	0.586	fus
guj_gujr	Gujarati	0.347	fus
hun_latn	Hungarian	0.739	agg
isl_latn	Icelandic	0.574	fus
ind_latn	Indonesian	0.708	agg
gle_latn	Irish	0.468	fus
jpn_jpan	Japanese	0.691	agg
kor_hang	Korean	0.692	agg
kmr_latn	Kurdish	0.202	fus
pes_arab	Persian	0.345	fus
slv_latn	Slovenian	0.650	fus
spa_latn	Spanish	0.592	fus
tam_taml	Tamil	0.435	agg
tur_latn	Turkish	0.591	agg
urd_arab	Urdu	0.747	fus
zul_latn	Zulu	0.541	agg

linear regression between the variables, but found no significant correlation ($F(1, 13)=0.323$, $p=0.580$).

Discussion

One possible explanation for this result is that words in agglutinative languages are on average being segmented into more tokens, making it more likely that a token boundary will fall on a morpheme boundary. This in turn could be driven by word length, as agglutinative languages tend to have longer words. It could also be due to a higher number of token boundaries per word (fertility), as higher fertility means that as there are more token boundaries, it becomes more likely that one of the token boundaries would fall on a morpheme boundary due to chance. Upon analysis, we found that agglutinative languages indeed had longer words ($t(29{,}923)=18.222$, $p<0.001$) and more tokens per word ($t(37{,}375)=34.27$, $p<0.001$). We fit a linear regression with number of tokens per word, word length in characters, and morphological types as predictors for MorphScore. We found that fertility and word length are both negatively correlated with MorphScore ($\chi^2(1)=61.457$, $p<0.001$; $\chi^2(1)=364.03$, $p<0.001$; respectively); however, the effect sizes were extremely small with an adjusted $R^2 = 0.021$. Given these small effects, longer words or higher fertility cannot explain the greater than 20% higher MorphScores for agglutinative languages.

In order to mitigate concern about the choice to exclude one-token words from the calculation of MorphScore, we also calculate MorphScore such that a one-token word is counted as correct. Agglutinative languages still had higher MorphScores than fusional languages ($t(18.874) = 2.393$, $p = 0.027$). Furthermore, we found no

difference in the absolute number of one-token words (t(19.867) = -0.768, p = 0.452) nor in the proportion of one-token words (t(17.014) = -0.577, p =0.572) between agglutinative and fusional languages.

These results are inconsistent with Hypothesis 1; morphological tokenizer alignment (as measured by MorphScore) is higher for agglutinative languages rather than lower, and this effect cannot be explained by higher fertility or longer word length.

4.1.6 H2: Tokenization Quality

We next evaluate whether tokenization quality can explain the performance gap between agglutinative and fusional languages. We use two metrics of tokenization quality: compression and Rényi entropy. To achieve sufficient statistical power, we use the same tokenizers as the previous section, but add all the languages from Chang et al. (2023a) with FLORES datasets and for which we have morphological type labels, for a total of 63 languages. Perplexities for each tokenizer come from Chang et al. (2023a).

Compression

We use corpus token count (CTC; also known as sequence length) as our measure of compression. CTC (Schmidt et al., 2024) is the number of tokens it takes to encode a text. Lower CTC indicates better compression, which is thought to have various effects on performance, cost, and inference time. If a tokenizer encodes a given text with more tokens, this will mean more sequences in order to pass the text

through a language model. Each sequence, thus, will contain less information. This leads to higher training cost and slower inference (Song et al., 2021; Petrov et al., 2023; Yamaguchi et al., 2024) and worse model performance (Gallé, 2019; Liang et al., 2023; Goldman et al., 2024).

We calculate CTC based on FLORES-200 (NLLB Team et al., 2022) by encoding the text for each language with its respective tokenizer and counting the sequence length, not including beginning- and end-of-sequence tokens. FLORES offers parallel texts for each language, meaning that each text contains the same content, and sequence lengths should be comparable between languages.

Rényi entropy

Rényi entropy has been proposed as a metric of tokenization quality, as it measures the distribution of token frequencies over the tokenizer vocabulary, penalizing low- and high-frequency tokens. It has been shown to correlate with downstream performance (Zouhar et al., 2023).

Rényi entropy might also capture undesirable tokenizer properties that could be causing the performance gap. Agglutinative languages have longer words (Fenk-Oczlon and Fenk, 1999; Berg et al., 2022) and more unique word forms (Sandra, 1994). This means that a tokenizer with a fixed vocabulary size will necessarily use shorter tokens on average for an agglutinative language than for a fusional language[4].

[4]As there has not been previous empirical evidence to support this point, we test this. We use the same tokenizers as in the previous section and tokenize all the FLORES datasets for which we have corresponding monolingual tokenizers. We then calculate mean token length for the FLORES dataset. The mean token length for fusional languages was 2.92 characters and the mean token length for agglutinative languages was 3.25. This difference is statistically significant ($t(68.36) = 3.236$, $p = 0.002$).

Shorter tokens will have higher frequencies on average (Berg et al., 2022), and these tokens will carry less information, as the meaning of a word is distributed over more tokens.

We calculate Rényi entropy from the FLORES dataset for each language using `tokenization-scorer`[5] (Zouhar et al., 2023) with the recommended setting (α=2.5).

Results

Agglutinative languages have higher CTC (worse compression) than fusional languages (t(85.944)=2.507, p=0.014). On average, their sequences are 3.5% longer. However, there is no correlation between CTC and perplexity (linear regression; F(1, 190)=2.05, p=0.154). This indicates that compression, at least measured in this way, does not explain the performance gap.

There is also a difference in Rényi entropy between agglutinative and fusional languages. Agglutinative languages have worse (higher) Rényi entropy (M=0.547) than fusional languages (M=0.488; t(150.53)=5.168, p<0.001).

In order to test whether Rényi entropy can help explain the performance gap, conduct a Likelihood Ratio Test comparing two linear mixed effects models. The full model predicts perplexity from morphological type, Rényi entropy, and CTC as fixed effects, with model size as a random intercept. We then fit a reduced model, removing morphological type as a fixed effect. We compare the models with an ANOVA and find that morphological type explains additional variance above and beyond the other predictors (χ^2(3)=29.464, p<0.001). This indicates that Rényi

[5] `https://github.com/zouharvi/tokenization-scorer`

entropy does not explain all of the variance significantly explained by morphological type. A variance partitioning analysis using the `partR2` package in R (**?**) produces an R^2 for morphological type of 0.100 and for Rényi entropy of 0.030, while the full model R^2 is 0.144. Therefore the vast majority of the variance is still explained by morphological type. This suggests that Rényi entropy could explain only a small part of the performance gap.

Discussion

These results are inconsistent with the hypothesis that tokenizer compression explains poorer language modeling performance for agglutinative languages. Other results, e.g. Deletang et al. (2024); Schmidt et al. (2024), also show a lack of relationship between compression and language model performance. While compression indicates how much information can be represented in a fixed sequence length, the effect of compression may be outweighed by other features of a particular tokenizer, or language models may be able to overcome suboptimal tokenization. This is an area for further research, as it remains unclear what the best criteria are for intrinsic evaluation of tokenizers (Zouhar et al., 2023; Chizhov et al., 2024).

4.1.7 H3: Data Measurement Disparities

The final hypothesis for the performance gap is disparities in training data.

The monolingual models used in §4.1.4 and §4.1.6 were designed to be trained on comparable amounts of training data with comparable tokenizers (Chang et al., 2023a). Nevertheless, there are differences in performance between languages. Chang

et al. (2024) trained a similar suite of models (the Goldfish models), taking into account the byte premiums for each language.

Byte premiums (Arnett et al., 2024a) are the ratio of the number of bytes it takes to represent a content-matched text in different languages. For example, a text in a language with a byte premium of 3 relative to English will be three times larger in bytes than the content-matched English text file. One of the major contributors to byte premiums is the writing system used by a language. Latin characters are represented with a single byte in UTF-8 encoding. In the most extreme cases, characters for scripts like Khmer take three bytes per character, not including diacritics. As a result, some languages have byte premiums of up to 5 relative to English.

This has implications for many things, including how much text tokenizers are trained on. Most training data can be measured in number of tokens, but this is not the case for tokenizer training data, as the tokenizer has not been trained yet. The Goldfish tokenizers and models are trained on byte-premium-scaled text quantities, which was designed to reduce the effects of the data measurement disparities between languages.

In this section, we test whether taking byte premiums into account can reduce or completely eliminate the performance gap. We annotate 154 languages for morphological type and use the same procedure as in §4.1.4 to test for the performance gap with the Goldfish models.

Results

The Goldfish models exhibit numerically higher perplexity for agglutinative (M=143.62) than fusional languages (M=132.63), but this difference is not statistically significant (t(137.36)=1.180, p=0.077). Therefore, after taking byte premiums into effect, the Goldfish models do not exhibit the same performance gap that was demonstrated in previous research and in Section 4.1.4 above.

We tested whether there was a relationship between byte premium and morphological type, and found that there was a marginally significant difference between byte premiums for agglutinative and fusional languages (t-test; t(157.9)=1.960, p=0.0518).

Discussion

The results show that after taking into account byte premiums, there is no difference in performance according to morphological typology. Thus, accounting for byte premiums by scaling training data reduces most of the variance previously accounted for by morphological type. This suggests, therefore, that differences that seemed to be driven by morphological typology are actually being driven by disparities in dataset size measurement.

4.2 Discussion

We find that byte premiums explain the largest portion of the performance gap, which means that cross-lingual differences in text encoding size can explain

these particular, previously documented performance differences. The results do not support the idea that some languages are harder to model than others, but it does seem that languages need to be treated differently, e.g. by scaling data quantities. This result can be used to inform how much data should be used to train tokenizers and language models, especially in low-resource or multilingual settings. By not taking byte premiums into account, we may be disadvantaging languages which are historically under-represented in the field, even when resources for them do exist.

While these results may be surprising based on the NLP literature, these results are consistent with evidence from language acquisition work, which has not shown any cross-linguistic differences in learnability of languages. These results are unsurprising from an empirical perspective, as work in Linguistics and NLP has consistently shown that more data will always facilitate better learning, irrespective of the complexity of the language.

There do seem to be limits to the learnability of linguistic systems. There are some language systems that linguistic theory predicts are impossible for humans to learn. Recent work has shown that language models are less successful at learning those languages, compared to existing and possible linguistic systems (Kallini et al., 2024). Therefore, we do not predict that these results hold for systems that are more complex than any attested natural language.

The results relating to Rényi entropy do suggest that there may be differences in tokenization which could be affecting performance, however more work is needed on this topic.

4.3 Conclusion

This paper first presented new evidence consistent with a performance gap between languages of different morphological types. We presented and tested three hypotheses as to the cause(s) for this performance gap: morphological alignment of the tokenizer, tokenization quality, and measurement disparities of dataset size. We found that while there was evidence that tokenization quality (as measured by Rényi entropy) plays a small role, dataset size seems to explain a large portion of the performance gap. After scaling training data according to byte premiums – a measure of how many bytes it takes to represent text in different languages – the performance gap goes away.

To do this work, we also created MorphScore, which is an evaluation method that can be used to evaluate the morphological alignment of tokenizers. We release the datasets needed to evaluate MorphScore in 22 languages: `https://github.com/catherinearnett/morphscore`.

These results raise questions about other unintended differences in the way languages are treated that could lead to differences in performance between languages. This is a critical issue for achieving language-general NLP systems and making language models perform equitably. While it does not seem that morphological typology is the primary reason for the observed performance gap, the initial observation led to greater understanding of crosslinguistic NLP. It is important to keep evaluating the dimensions along which languages vary and considering whether language technologies, such as LLMs, introduce inequalities between languages. We have yet to fully understand all the ways in which English-centric practices in NLP

may have impeded progress for language models in other languages.

Limitations

For all of the analyses, we were limited by the number of languages for which we had morphological type annotations. These annotations are time-consuming and are themselves limited by the resources available, namely grammars and linguistic descriptions. The number of languages in the MorphScore analysis is even more limited. Having more annotations and datasets included in this work would make the analyses more reliable. This is an important place for expansion in future work.

In the MorphScore datasets, we were also limited by the type of existing data. There were differences in domain and breadth in the Universal Dependencies and UniMorph datasets for each language. There are also different numbers of items in each dataset for each language. This means some languages will have more diversity among the items and there will be more statistical power than others, therefore the treatment of each language was not the same, which could introduce uncontrolled variance. Additionally, the morpheme boundaries that are annotated for different languages was not consistent. Some boundaries were inflectional and some were derivational. If there existed large datasets of both inflectional and derivational morphologically annotated words in a wide range of languages, this would have improved the robustness of the MorphScore results.

Finally, because the annotations in UD and UniMorph chose only one boundary (or, if there were multiple boundaries, we chose one), we can only evaluate

whether the token boundaries align with the morpheme boundary we chose. We did this to limit confounds, as all but one dataset had one annotated boundary per word. Additionally, agglutinative languages would have more morpheme boundaries per word, which could skew results. However, there was no controlled selection process for which morpheme boundary was used for the MorphScore analysis, therefore this could have also affected results.

In the analysis of the Goldfish models, the evidence that byte premiums account for the performance gap is supported by a marginally significant difference between byte premiums according to their morphological type. It is possible that with an even larger sample of languages, the effect would instead meet the standard threshold for significance. We argue that in conjunction with the other results, it still demonstrates that taking byte premiums into account significantly reduces the performance gap.

Acknowledgments

This chapter is a pre-print of a paper published in the proceedings of The 31st International Conference on Computational Linguistics (COLING 2025).

Part I Discussion

Part I of the dissertation discusses two areas where crosslinguistic differences should be considered in NLP: training data and tokenizers.

With respect to training data, in Chapter 2, I show that encoding standards introduce crosslinguistic disparities in dataset size measurement due to differences in word length and writing systems. In Chapter 4, I show that these differences are driving crosslinguistic differences in language model performance. Accounting for these differences accounts for most of the variance that was previously attributed to typological differences.

Relating to tokenizers, in Chapter 3, I show some evidence that morphologically aligned tokenization may not be necessary for accurate performance on a linguistic task, subject-verb agreement. In Chapter 4, I show that morphologically aligned tokenization does not seem to be linked to language model performance. Together, these results challenge the view that morphological segmentation represents the ideal tokenization of a given language.

Here, I discuss some overarching questions and discussions that these results contribute to.

Language Bias in Deep Learning In Chapter 4, I show that crosslinguistic differences in performance can be explained by byte premiums and the dataset size measurement disparities they introduce. This substantiates the criticisms brought forth by Bender (2011); however, it is not necessarily the neural architectures or tokenization algorithms that are responsible for these biases. Instead, I show that the most widely used encoding standard, UTF-8, is responsible for introducing a significant amount of bias.

In some ways this is bodes well for improving language equity in NLP. It does seem that the models that have been developed for English can generally be applied to other languages. This is not particularly surprising, given that the largest and most commercially successful models are massively multilingual, i.e. being trained on data from many languages, and therefore has some pressure to work well for multiple languages. However, UTF-8 encoding is deeply embedded in nearly all digital applications and products, so overcoming inequities introduced by this encoding standard will likely have to operate within this encoding standard.

There are existing alternatives to UTF-8. UTF-16 is a similar encoding standard, in which characters are represented with either 2 or 4 bytes. Most characters are represented with 2 bytes, but rarer characters are represented by 4 bytes. Therefore, there would be more equity among the most frequent writing systems, but there would still be some writing systems largely represented with four-byte characters. UTF-32 is even more equitable. Using this encoding standard, all characters are represented with 4 bytes. This would be far more equitable, however it means that most languages would use many more bytes to represent a given text relative to

UTF-8 encoding. Additionally, this would not eliminate differences in the number of characters needed to represent a given amount of information across languages. Therefore, there would still be byte premium effects.

MYTE (Limisiewicz et al., 2024), which was described above in Chapter 2, is a possible solution to byte premium effects. The authors show that it reduces byte premiums across languages; however, as I discuss above, it requires supervised morphological segmentation datasets, which are not available for a lot of languages. Byte Latent Transformer (BLT) is a novel byte-level language model architecture, which dynamically groups bytes according to the entropy of the next byte (Pagnoni et al., 2024). Under this approach, the amount of compression changes based on the complexity of the text. The architecture was primarily trained and evaluated on English data, therefore it is unclear whether this could address byte premiums.

The main takeaway from these experiments is that differences in performance can be attributed to a lack of data, which is then exacerbated by byte premiums. Dataset creation, however needs to be with the consent and participation of the language community, especially in the case of low-resource languages. And not all data is the same. Low-resource languages often have less and lower-quality data, which is often full of errors and is culturally irrelevant (Kreutzer et al., 2022; Nigatu et al., 2023). All these factors must be considered to fully address the problem of data scarcity.

Does Tokenization Even Matter? In Chapter 3, I showed that for agreement tasks, morphologically aligned tokenization did not significantly impact performance. In Chapter 4, I show that tokenization does not seem to explain any crosslinguistic

variance in performance. So, does tokenization matter at all in terms of language model performance?

There have been multiple studies in which researchers have shown that models which differ only in tokenizer can have different performance levels (Ali et al., 2024; Schmidt et al., 2024). The differences are not that big, so it seems that language models are generally robust to apparently sub-optimal tokenization. Sub-optimal tokenization has even been intentionally added into model training, via BPE Dropout (Provilkov et al., 2020), which introduces slight variations in the tokenization algorithm during model training. This trains the model to handle segmentation errors better and also leads to better downstream performance.

To the extent that tokenizers impact model performance, it is unclear what features of a tokenizer affect performance. Compression has long thought to be one of the primary metrics of tokenizer quality linked to downstream performance, but recent results show that this is not the case. Schmidt et al. (2024) trained identical models, while manipulating only the tokenizer. They found that compression was not correlation with downstream performance. In that study, the authors did not find any feature that was predictive of better performance.

Apart from compression, proposed tokenizer evaluation metrics include proportion of word-initial tokens (Yehezkel and Pinter, 2023), Rényi efficiency (Zouhar et al., 2023), and proportion of continued words (i.e. non-word-initial tokens; Occiglot, 2024). But there has been no consensus about which features of a tokenizer are m (Zouhar et al., 2023; Chizhov et al., 2024).

Other features of tokenizers also seem important to consider. Land and Bar-

tolo (2024) show that many tokenizers contain tokens, which the model does not see very often during training. As a result, the embeddings for those tokens are not updated much and remain very close to their initialized embeddings, making them under-trained tokens. These tokens usually cause the model to hallucinate, which makes them generate text off-topic. These tokens can also cause models to circumvent safety guardrails and generate harmful text (Geiping et al., 2024).

It is unclear how tokenizers should be evaluated. This is a key area for future work in order to better determine the role of tokenization in language models and how best to optimize tokenizers.

Alternatives to the Dual-Route Hypothesis The two approaches that have been primarily discussed in the tokenization literature are the decomposition hypothesis and the dual-route encoding hypothesis. However, the results from these chapters also support a connectionist account (Rumelhart and McClelland, 1986), wherein a set of parameters may be tuned through exposure to represent a system in a distributed way. Language models are embody the connectionist account (Joanisse and McClelland, 2015) and show that it is possible to generate linguistically correct text through distributional information alone.

Unlike the dual-route account, the connectionist account treats all forms uniformly, without the need to determine regular and irregular forms, or frequent and infrequent tokens. This provides some advantage to the connectionist approach, as the delineations between those categories can be difficult to determine. The connectionist account not only captures observed statistical effects, but also the context-sensitive and usage-based phenomena, for which there is increasing empirical evidence

(Stevens and Plaut, 2022).

A connectionist approach to tokenization would not assume that tokenizers should generate linguistically meaningful or interpretable tokens. Instead, this approach would assume that there may be another optimal tokenization scheme that can be learned by the neural network, even if the tokenization scheme does not appear to be optimal. Adopting this view of tokenization could be helpful in shaping the approach to tokenization, in particular to the evaluation of tokenization, especially in light of the empirical evidence that challenges the rule-based, decomposition approach to tokenization.

This is consistent with other work in NLP, which finds that adding explicit linguistic information does not improve language model performance (Toraman et al., 2023; Jeon et al., 2023), as evidenced by the lack of widespread adoption of explicitly encoding linguistic knowledge.

Part II

Understanding Crosslingual Transfer through Structural Priming

In Part II, I seek to provide evidence that crosslingual transfer occurs because of shared multilingual representations. I seem to characterize when the models learn these representations, how they change over the course of training, and what the representations are actually representing. This is important for better understanding crosslingual transfer. In particular, it is important to identify the limitations of crosslingual transfer in order to improve the utility of this phenomenon.

As mentioned in Chapter 1, crosslingual transfer is the most widely used strategy for improving low-resource language performance. But training a language model in a multilingual model may lead to the curse of multilinguality (Conneau et al., 2020a; Chang et al., 2023a). Therefore, crosslingual transfer does not always lead to better performance. Without sufficient data or model capacity, a model may represent the high-resource "priority" languages quite well and the lower-resource languages quite poorly. The degree to which the lower-resource languages are harmed by these practices are difficult to assess first because even these practices may lead to the best model performance for a language, despite the language model not being able to reliably produce grammatical sentences in the target language; and second, because there is a severe lack of meaningful evaluations for many lower-resource languages.

Because the limitations of crosslingual transfer are not well understood, it remains the default approach to developing language models for languages other than English. As a result, crosslingual transfer is the justification for many NLP practitioners to continue to pursue an English-centric NLP, because many people believe that representations learned from English data can simply be transformed

into representations for another (often, lower-resource) language with very little input data from the target low-resource language. As a result, the vast majority of models are trained on English data, under the assumption that all natural language expressions can be perfectly represented in a language-neutral latent space and that these representations can be losslessly transformed between any two languages. But fine-grained social and cultural meaning, including culturally-specific connotations and context-dependent interpretations, is translatable.

And as English is often the source language for crosslingual transfer, the implicit assumption is that English can be used to build language-neutral representations. But English is not language-neutral, nor is any language. The task text cannot be generated in a language-neutral way. It is impossible – and, I argue, undesirable – to seek a text generation model that operates in a language-neutral space. This would result in text which is devoid of any real social or cultural significance.

But given the current practices, practitioners are implicitly choosing instead to operate in an English-centric linguistic and cultural space. One of the ways this can be seen is through cultural bias exhibited by language models. A recent study found that major commercial models, e.g. GPT-4, generate responses consistent with the social and cultural values of the US and Protestant Europe (Tao et al., 2024).

Data remains a primary obstacle to changes to these practices. Especially for particular domains and applications, it may be difficult or impossible to find training data from a desired domain in the target language. Academic articles are a concrete example of this. The vast majority of academic publishing occurs in English, therefore there is domain knowledge that is largely only available from English text

data.

The work in Part II of this dissertation contributes to the relatively small number of studies on crosslingual transfer and how it works. A better understanding of crosslingual transfer can help inform NLP practitioners on when and how to use multilingual settings to improve language model performance from lower-resource languages.

Chapter 5

Structural Priming Demonstrates Abstract Grammatical Representations in Multilingual Language Models

Abstract

Abstract grammatical knowledge—of parts of speech and grammatical patterns—is key to the capacity for linguistic generalization in humans. But how abstract is grammatical knowledge in large language models? In the human literature, compelling evidence for grammatical abstraction comes from structural priming. A sentence that shares the same grammatical structure as a preceding sentence is processed

and produced more readily. Because confounds exist when using stimuli in a single language, evidence of abstraction is even more compelling from crosslingual structural priming, where use of a syntactic structure in one language primes an analogous structure in another language. We measure crosslingual structural priming in large language models, comparing model behavior to human experimental results from eight crosslingual experiments covering six languages, and four monolingual structural priming experiments in three non-English languages. We find evidence for abstract monolingual and crosslingual grammatical representations in the models that function similarly to those found in humans. These results demonstrate that grammatical representations in multilingual language models are not only similar across languages, but they can causally influence text produced in different languages.

5.1 Introduction

What do language models learn about the structure of the languages they are trained on? Under both more traditional generative (Chomsky, 1965) and cognitively-inspired usage-based theories of language (Tomasello, 2003; Goldberg, 2006; Bybee, 2010), the key to generalizable natural language comprehension and production is the acquisition of grammatical structures that are sufficiently abstract to account for the full range of possible sentences in a language. In fact, both theoretical and experimental accounts of language suggest that grammatical representations are abstract enough to be shared across languages in both humans (Heydel and Murray, 2000; Hartsuiker et al., 2004; Schoonbaert et al., 2007) and language

models (Conneau et al., 2020b,a; Jones et al., 2021).

The strongest evidence for grammatical abstraction in humans comes from *structural priming*, a widely used and robust experimental paradigm. Structural priming is based on the hypothesis that grammatical structures may be activated during language processing. Priming then increases the likelihood of production or increased ease of processing of future sentences sharing the same grammatical structures (Bock, 1986; Ferreira and Bock, 2006; Pickering and Ferreira, 2008; Dell and Ferreira, 2016; Mahowald et al., 2016; Branigan and Pickering, 2017). For example, Bock (1986) finds that people are more likely to produce an active sentence (e.g. *one of the fans punched the referee*) than a passive sentence (e.g. *the referee was punched by one of the fans*) after another active sentence. This has been argued (Bock, 1986; Heydel and Murray, 2000; Pickering and Ferreira, 2008; Reitter et al., 2011; Mahowald et al., 2016; Branigan and Pickering, 2017) to demonstrate common abstractions generalized across all sentences with the same structure, regardless of content.

Researchers have found evidence that structural priming for sentences with the same structure occurs even when the two sentences are in different languages (Loebell and Bock, 2003; Hartsuiker et al., 2004; Schoonbaert et al., 2007; Shin and Christianson, 2009; Bernolet et al., 2013; van Gompel and Arai, 2018; Kotzochampou and Chondrogianni, 2022). This *crosslingual structural priming* takes abstraction one step further. First, it avoids any possible confounding effects of lexical repetition and lexical priming of individual words—within a given language, sentences with the same structure often share function words (for discussion, see Sinclair et al., 2022).

More fundamentally, crosslingual structural priming represents an extra degree of grammatical abstraction not just within a language, but across languages.

We apply this same logic to language models in the present study. While several previous studies have explored structural priming in language models (Prasad et al., 2019; Sinclair et al., 2022; Frank, 2021; Li et al., 2022; Choi and Park, 2022), to the best of our knowledge, this is the first to look at crosslingual structural priming in Transformer language models. We replicate eight human psycholinguistic studies, investigating structural priming in English, Dutch (Schoonbaert et al., 2007; Bernolet et al., 2013), Spanish (Hartsuiker et al., 2004), German (Loebell and Bock, 2003), Greek (Kotzochampou and Chondrogianni, 2022), Polish (Fleischer et al., 2012), and Mandarin (Cai et al., 2012). We find priming effects in the majority of the crosslingual studies and all of the monolingual studies, which we argue supports the claim that multilingual models have shared grammatical representations across languages that play a functional role in language generation.

5.2 Background

Structural priming effects have been observed in humans both within a given language (Bock, 1986; Ferreira and Bock, 2006; Pickering and Ferreira, 2008; Dell and Ferreira, 2016; Mahowald et al., 2016; Branigan and Pickering, 2017) and crosslingually (Loebell and Bock, 2003; Hartsuiker et al., 2004; Schoonbaert et al., 2007; Shin and Christianson, 2009; Bernolet et al., 2013; van Gompel and Arai, 2018; Kotzochampou and Chondrogianni, 2022). In language models, previous work has

demonstrated structural priming effects in English (Prasad et al., 2019; Sinclair et al., 2022; Choi and Park, 2022), and initial results have found priming effects between English and Dutch in LSTM language models (Frank, 2021). As these studies argue, the structural priming approach avoids several possible assumptions and confounds found in previous work investigating abstraction in grammatical learning. For example, differences in language model probabilities for individual grammatical vs. ungrammatical sentences may not imply that the models have formed abstract grammatical representations that generalize across sentences (Sinclair et al., 2022); other approaches involving probing (e.g. Hewitt and Manning, 2019; Chi et al., 2020) often do not test whether the internal model states are causally involved in the text predicted or generated by the model (Voita and Titov, 2020; Sinclair et al., 2022). The structural priming paradigm allows researchers to evaluate whether grammatical representations generalize across sentences in language models, and whether these representations causally influence model-generated text. Furthermore, structural priming is agnostic to the specific language model architecture and does not rely on direct access to internal model states.

However, the structural priming paradigm has not been applied to modern multilingual language models. Previous work has demonstrated that multilingual language models encode grammatical features in shared subspaces across languages (Chi et al., 2020; Chang et al., 2022; de Varda and Marelli, 2023), largely relying on probing methods that do not establish causal effects on model predictions. Crosslingual structural priming would provide evidence that the abstract grammatical representations shared across languages in the models have causal effects on

model-generated text. It would also afford a comparison between grammatical representations in multilingual language models and human bilinguals. These shared grammatical representations may help explain crosslingual transfer abilities in multilingual models, where tasks learned in one language can be transferred to another (Artetxe et al., 2020b; Conneau et al., 2020a,b; K et al., 2020; Goyal et al., 2021; Ogueji et al., 2021; Armengol-Estapé et al., 2021, 2022; Blevins and Zettlemoyer, 2022; Chai et al., 2022; Muennighoff et al., 2023; Wu et al., 2022; Guarasci et al., 2022; Eronen et al., 2023).

Thus, this study presents what is to our knowledge the first experiment testing for crosslingual structural priming in Transformer language models. The findings broadly replicate human structural priming results: higher probabilities for sentences that share grammatical structure with prime sentences both within and across languages.

5.3 Method

We test multilingual language models for structural priming using the stimuli from eight crosslingual and four monolingual priming studies in humans. Individual studies are described in §5.4.

5.3.1 Materials

All replicated studies have open access stimuli with prime sentences for different constructions (§5.3.3). Where target sentences are not provided (because

participant responses were manually coded by the experimenters), we reconstruct target sentences and verify them with native speakers.

5.3.2 Language Models

We test structural priming in XGLM 4.5B (Lin et al., 2022), a multilingual autoregressive Transformer trained on data from all languages we study in this paper, namely, English, Dutch, Spanish, German, Greek, Polish, and Mandarin. To the best of our knowledge, this is the only available pre-trained (and not fine-tuned) autoregressive language model trained on all the aforementioned languages. To avoid drawing any conclusions based on the idiosyncrasies of a single language model, we also test a number of other multilingual language models trained on most of these languages, namely the other XGLM models, i.e., 564M, 1.7B, 2.9B, and 7.5B, which are trained on all the languages except for Dutch and Polish; and PolyLM 1.7B and 13B (Wei et al., 2023), which are trained on all the languages except for Greek.

5.3.3 Grammatical Alternations Tested

We focus on structural priming for the three alternations primarily used in existing human studies.

Dative Alternation (DO/PO) Some languages permit multiple orders of the direct and indirect objects in sentences. In PO (prepositional object) constructions, e.g., *the chef gives a hat to the swimmer* (Schoonbaert et al., 2007), the direct object *a hat* immediately follows the verb and the indirect object is introduced with the

prepositional phrase *to the swimmer*. In DO (double object) constructions, e.g., *the chef gives the swimmer a hat*, the indirect object *the swimmer* appears before the direct object *a hat* and neither is introduced by a preposition. Researchers compare the proportion of DO or PO sentences produced by experimental participants following a DO or PO prime.

Active/Passive In active sentences the syntactic subject is the agent of the action, while in passive sentences the syntactic subject is the patient or theme of the action. E.g., *the taxi chases the truck* is active, and *the truck is chased by the taxi* is passive (Hartsuiker et al., 2004). Researchers compare the proportion of active or passive sentences produced by experimental participants following an active or passive prime.

Of-/S-Genitive *Of*- and *S*-Genitives represent two different ways of expressing possessive meaning. In an *of*-genitive, the possessed thing is followed by a preposition such as *of* and then the possessor, e.g., *the scarf of the boy is yellow*. In *s*-genitives in the languages we analyze (English and Dutch), the possessor is followed by a word or an attached morpheme such as *'s* which is then followed by the possessed thing, e.g., *the boy's scarf is yellow* (Bernolet et al., 2013). Researchers compare the proportion of *of*-genitive or *s*-genitive sentences produced by experimental participants following an *of*-genitive or *s*-genitive prime.

5.3.4 Testing Structural Priming in Models

In human studies, researchers test for structural priming by comparing the proportion of sentences (targets) of given types produced following primes of different

types. Analogously, for each experimental item, we prompt the language model with the prime sentence and compute the normalized probabilities of each of the two target sentences. We illustrate our approach to computing these normalized probabilities below.

First, consider the example dative alternation stimulus sentences from Schoonbaert et al. (2007):

(1) a. **DO prime:** The cowboy shows the pirate an apple.

b. **PO prime:** The cowboy shows an apple to the pirate.

c. **DO target:** The chef gives the swimmer a hat.

d. **PO target:** The chef gives a hat to the swimmer.

We can use language models to calculate the probability of each target following each prime by taking the product of the conditional probabilities of all tokens in the target sentence given the prime sentence and all preceding tokens in the target sentence. In practice, these probabilities are very small, but for illustrative purposes, we can imagine these have the following probabilities:

P(PO Target | DO Prime) = 0.03

P(DO Target | DO Prime) = 0.02

P(PO Target | PO Prime) = 0.04

P(DO Target | PO Prime) = 0.01

We then normalize these probabilities by calculating the conditional probability of each target sentence given that the model response is one of the two target

sentences, as shown below.

$$P_N(\text{PO} \mid \text{DO}) = 0.03/(0.03{+}0.02) = 0.60$$

$$P_N(\text{DO} \mid \text{DO}) = 0.02/(0.03{+}0.02) = 0.40$$

$$P_N(\text{PO} \mid \text{PO}) = 0.04/(0.04{+}0.01) = 0.80$$

$$P_N(\text{DO} \mid \text{PO}) = 0.01/(0.04{+}0.01) = 0.20$$

Because the normalized probabilities of the two targets following a given prime sum to one, we only consider the probabilities for one target type in our analyses (comparing over the two different prime types). For example, to test for a priming effect, we could either compare the difference between $P_N(\text{PO} \mid \text{PO})$ and $P_N(\text{PO} \mid \text{DO})$ or the difference between $P_N(\text{DO} \mid \text{PO})$ and $P_N(\text{DO} \mid \text{DO})$. We follow the original human studies in the choice of which target construction to plot and test.

We run statistical analyses, testing whether effects are significant for each language model on each set of stimuli. To do this, we construct a linear mixed-effects model predicting the target sentence probability (e.g. probability of a PO sentence) for each item. We include a random intercept for experimental item, and we test whether prime type (e.g. DO vs. PO) significantly predicts target structure probability. All reported p-values are corrected for multiple comparisons by controlling for false discovery rate (Benjamini and Hochberg, 1995). All stimuli, data, code, and statistical analyses are provided at `https://osf.io/2vjw6/`.

5.4 Results

In reporting whether the structural priming effects from human experiments replicate in XGLM language models, we primarily consider the direction of each effect in the language models (e.g. whether PO constructions are more likely after PO vs. DO primes) rather than effect sizes or raw probabilities. The mean of the relative probabilities assigned by language models to the different constructions in each condition may not be directly comparable to human probabilities of production. Humans are sensitive to contextual cues that may not be available to language models; notably, in these tasks, humans are presented with pictures corresponding to events in the structural priming paradigm. Furthermore, construction probabilities in language models may be biased by the frequency of related constructions in any of the many languages on which the models are trained. Thus, we focus only on whether the language models replicate the direction of the principal effect in each human study.

5.4.1 Crosslingual Structural Priming

We test whether eight human crosslingual structural priming studies replicate in language models. These studies cover structural priming between English and Dutch (Schoonbaert et al., 2007; Bernolet et al., 2013), Spanish (Hartsuiker et al., 2004), German (Loebell and Bock, 2003), Greek (Kotzochampou and Chondrogianni, 2022), and Polish (Fleischer et al., 2012). For each experiment, we show the original human probabilities and the normalized probabilities calculated using each language

Figure 5.1: Human and language model results for crosslingual structural priming experiments.

model, as well as whether there is a significant priming effect (Figure 5.1). The full statistical results are reported in section C.2.

Schoonbaert et al. (2007): Dutch→English

Schoonbaert et al. (2007) prime 32 Dutch-English bilinguals with 192 Dutch sentences with either prepositional (PO) or dative object (DO) constructions. Schoonbaert et al. (2007) find that experimental participants produce more PO sentences when primed with a PO sentence than when primed with a DO sentence (see Figure 5.1A). We see the same pattern with nearly all the language models (Figure 5.1A). With the exception of XGLM 1.7B, where the effect is only marginally significant after

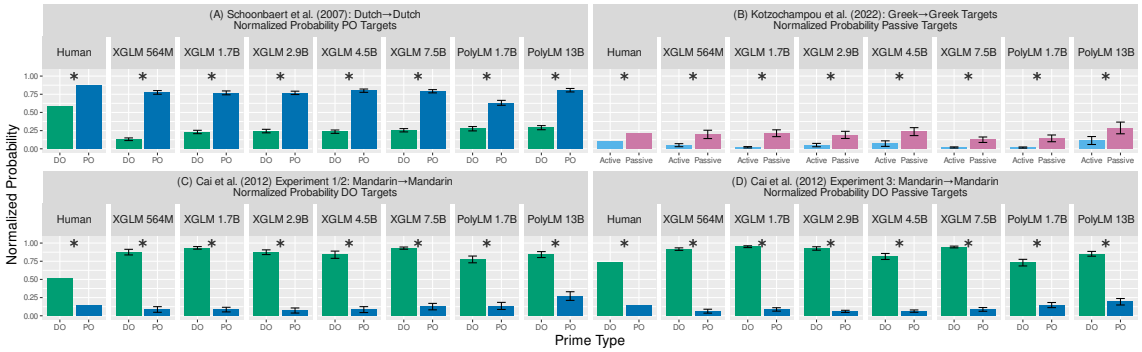

Figure 5.2: Human and language model results for within-language structural priming experiments.

correction for multiple comparisons, all language models predict English PO targets to be significantly more likely when they follow Dutch PO primes than when they follow Dutch DO primes.

Schoonbaert et al. (2007): English→Dutch

Schoonbaert et al. (2007) also observe DO/PO structural priming from English to Dutch (32 participants; 192 primes). As seen in Figure 5.1B, all language models show a significant priming effect.

Bernolet et al. (2013): Dutch→English

Bernolet et al. (2013) conduct a Dutch→English structural priming experiment with 24 Dutch-English bilinguals on 192 prime sentences, and they find that the production of *s*-genitives is significantly more likely after an *s*-genitive prime than after an *of*-genitive prime. We also observe this in all of the language models, as seen in Figure 5.1C.

Hartsuiker et al. (2004): Spanish→English

Hartsuiker et al. (2004) investigate Spanish→English structural priming with 24 Spanish-English bilinguals on 128 prime sentences, finding a significantly higher proportion of passive responses after passive primes than active primes. As shown in Figure 5.1D, this effect is replicated by XGLM 564M, 2.9B, and 7.5B as well as PolyLM 13B, with XGLM 4.5B showing a marginal effect ($p = 0.0565$).

Loebell and Bock (2003): German→English

Loebell and Bock (2003) find a small but significant priming effect of dative alternation (DO/PO) from German to English with 48 German-English bilinguals on 32 prime sentences. As can be seen in Figure 5.1E, while all language models show a numerical effect in the correct direction, the effect is only significant for XGLM 7.5B.

Loebell and Bock (2003): English→German

Loebell and Bock (2003) also test 48 German-English bilinguals for a dative alternation (DO/PO) priming effect from English primes to German targets (32 prime sentences), finding a small but significant priming effect. As we show in Figure 5.1F, the models are relatively varied in direction of numerical difference. However, only XGLM 2.9B and PolyLM 13B display a significant effect, and in both cases the effect is in the same direction as that found with human participants.

Kotzochampou and Chondrogianni (2022): Greek→English

Kotzochampou and Chondrogianni (2022) find active/passive priming from Greek to English in 25 Greek-English bilinguals. Participants are more likely to produce passive responses after passive primes (48 prime sentences). As shown in Figure 5.1G), all XGLMs display this effect, while the PolyLMs, which are not trained on Greek, do not.

Fleischer et al. (2012): Polish→English

Similarly, Fleischer et al. (2012) find active/passive priming from Polish to English in 24 Polish-English bilinguals on 64 prime sentences. As we see in Figure 5.1H, while all models show a numerical difference in the correct direction, the effect is only significant for XGLM 564M, 2.9B, and 7.5B, and for PolyLM 1.7B.

5.4.2 Monolingual Structural Priming

In the previous section, we found crosslingual priming effects in language models for the majority of crosslingual priming studies in humans. However, six of the eight studies have English target sentences. Our results up to this point primarily show an effect of structural priming on English targets. While both previous work (Sinclair et al., 2022) and our results in §5.4.1 may indeed demonstrate the effects of abstract grammatical representations on generated text in English, we should not assume that such effects can reliably be observed for other languages. Thus, we test whether multilingual language models exhibit within-language structural priming effects comparable to those found in human studies for Dutch (Schoonbaert

et al., 2007), Greek (Kotzochampou and Chondrogianni, 2022), and two studies in Mandarin (Cai et al., 2012).

Schoonbaert et al. (2007): Dutch→Dutch

Using Dutch prime and target sentences (192 primes), Schoonbaert et al. (2007) find that Dutch-English bilinguals (N=32) produce PO sentences at a higher rate when primed by a PO sentence compared to a DO sentence. As we see in Figure 5.2A, all language models display this effect.

Kotzochampou and Chondrogianni (2022): Greek→Greek

In their Greek→Greek priming experiment, Kotzochampou and Chondrogianni (2022) find an active/passive priming effect in native Greek speakers (N=25) using 48 primes. As shown in Figure 5.2B, this effect is replicated by all language models.

Cai et al. (2012): Mandarin→Mandarin

Using two separate sets of stimuli, Cai et al. (2012) find within-language DO/PO priming effects in native Mandarin speakers (N=28, N=24).[1] As seen in Figure 5.2C and 5.2D, all language models show significant effects for both sets of stimuli (48 prime sentences in their Experiments 1 and 2, and 68 prime sentences in their Experiment 3).

[1]The original study tests the effect of variants of DO/PO primes (topicalized DO/PO and *Ba*-DO; see Cai et al., 2012). To unify our analyses across studies, we only look at structural priming following the canonical DO and PO primes used in both Experiments 1 and 2 of the original study, as well as those used in Experiment 3.

5.4.3 Further Tests of Structural Priming

We have now observed within-language structural priming in multilingual language models for languages other than English. In §5.4.1, we found robust English → Dutch structural priming (Schoonbaert et al., 2007) but only limited priming effects for targets in German. Although there are no human results for the non-English targets in the other studies in §5.4.1, we can still evaluate crosslingual structural priming with non-English targets in the language models by switching the prime and target sentences in the stimuli. Specifically, we test structural priming from English to Dutch (Bernolet et al., 2013), Spanish (Hartsuiker et al., 2004), Polish (Fleischer et al., 2012), and Greek (Kotzochampou and Chondrogianni, 2022).

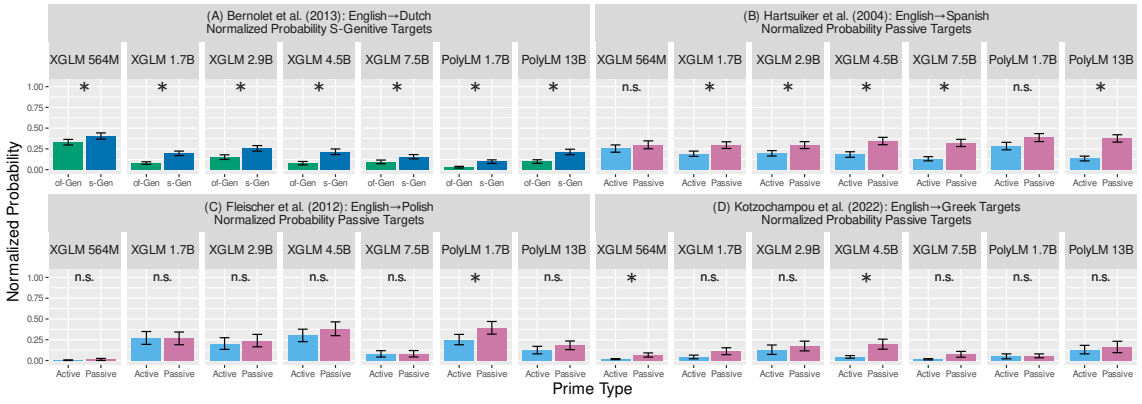

Figure 5.3: Language model results for structural priming experiments with no human baseline.

All models show a significant effect on the reversed Bernolet et al. (2013) stimuli (Figure 5.3A; English→Dutch), and all models but PolyLM 1.7B show the same for the reversed Hartsuiker et al. (2004) stimuli (Figure 5.3B; English→Spanish). The other results are less clear-cut. While XGLM 564M, 2.9B, and 4.5B and the

PolyLMs show a numerical effect in the correct direction for the reversed Fleischer et al. (2012) stimuli (English→Polish; Figure 5.3C), only PolyLM 1.7B shows a significant effect. For the reversed Kotzochampou and Chondrogianni (2022) stimuli (English→Greek; Figure 5.3D), all the XGLMs and PolyLM 13B show a numerical tendency in the correct direction, but only XGLM 564M and 4.5B show a significant effect.

5.5 Discussion

We find structural priming effects in at least one language model on each set of stimuli (correcting for multiple comparisons). Moreover, we observe a significant effect in all models with the monolingual stimuli, and in the majority of the models for 8 of the 12 crosslingual stimuli. In line with previous work (Hewitt and Manning, 2019; Chi et al., 2020), this supports the claim that language models learn generalized, abstract, and multilingual representations of grammatical structure. Our results further suggest that these shared grammatical representations are causally linked to model output.

5.5.1 Differences between models

In some ways, we see expected patterns across models. For example, for the XGLMs trained on 30 languages (XGLM 564M, 1.7B, 2.9B, and 7.5B), the larger models tend to display larger effect sizes than the smaller models, in line with the idea that model performance can scale with number of parameters (Brown et al., 2020;

Kaplan et al., 2020; Rae et al., 2022; Hoffmann et al., 2022; Touvron et al., 2023a). Additionally, the PolyLMs, which are not trained on Greek, do not show crosslingual structural priming for Greek (neither Greek→English nor English→Greek).

On the other hand, one surprising finding is that despite not being trained on Greek, the PolyLMs are able to successfully model monolingual structural priming in Greek. The most likely explanation for this is what Sinclair et al. (2022) refer to as 'lexical overlap'—the overlap of function words between primes and targets substantially boosts structural priming effects. In the same way that humans find it easier to process words that have recently been mentioned (Rugg, 1985, 1990; Van Petten et al., 1991; Besson et al., 1992; Mitchell et al., 1993; Rommers and Federmeier, 2018), language models may predict that previously-mentioned words are more likely to occur again (a familiar phenomenon in the case of repeated text loops; see Holtzman et al., 2020; See et al., 2019; Xu et al., 2022a) even if they are not trained on the words explicitly. This would explain the results for the Kotzochampou and Chondrogianni (2022) stimuli, as the Greek passive stimuli always include the preposition από.

Such an explanation could also account for the performance of XGLM 564M, 1.7B, 2.9B, and 7.5B on the Dutch and Polish stimuli. Despite not being intentionally trained on Dutch or Polish, we see robust crosslingual Dutch→English and English→Dutch structural priming, as well as Polish→English structural priming, in three of these models. However, as discussed previously, crosslingual structural priming avoids the possible confound of lexical overlap. For these results, therefore, a more likely explanation is language contamination. In contemporaneous work, we

find that training on fewer than 1M tokens in a second language is sufficient for structural priming effects to emerge (Arnett et al., 2023); our estimates of the amount of language contamination in XGLM 564M, 1.7B, 2.9B, and 7.5B range from 1.77M tokens of Dutch and 1.46M tokens of Polish at the most conservative to 152.5M and 33.4M tokens respectively at the most lenient (see section C.1).

The smaller amount of Polish contamination, as well as the fact that Polish is less closely related to English, may explain the less consistent Polish→English structural priming effects and the virtually non-existent English→Polish effects in these models, but as will be discussed in §5.5.2, there may be other reasons for this latter pattern.

5.5.2 Null Effects and Asymmetries

More theoretically interesting is the question of why some language models fail to display crosslingual structural priming on some sets of stimuli, even when trained on both languages. For example, in the Loebell and Bock (2003) replications, only XGLM 7.5B shows a significant effect of German→English structural priming, and only XGLM 2.9B and PolyLM 13B show a significant effect of English→German structural priming. This may be due to the grammatical structures used in the stimuli (DO/PO). While the original study does find crosslingual structural priming effects, the effect sizes are small; the authors suggest that this may partly be because "the prepositional form is used more restrictively in German" (Loebell and Bock, 2003, p. 807).

We also see an asymmetry in the crosslingual structural priming effects be-

tween some languages. While the effects in the Dutch→English (Bernolet et al., 2013) and Spanish→English (Hartsuiker et al., 2004) studies mostly remain when the direction of the languages is reversed, this is not the case for the Polish→English (Fleischer et al., 2012) and Greek→English (Kotzochampou and Chondrogianni, 2022) results. This may be due to the smaller quantity of training data for Polish and Greek compared to Spanish in XGLM. While XGLM is only trained on slightly more Dutch than Polish, Dutch is also more similar to English in terms of its lexicon and morphosyntax, so it may benefit from more effective crosslingual transfer (Conneau et al., 2020b; Gerz et al., 2018a; Guarasci et al., 2022; Winata et al., 2022; Ahuja et al., 2022; Oladipo et al., 2022; Eronen et al., 2023).

If it is indeed the case that structural priming effects in language models are weaker when the target language is less trained on, this would contrast with human studies, where crosslingual structural priming appears most reliable when the prime is in participants' native or primary language (L1) and the target is in their second language (L2). The reverse case often results in smaller effect sizes (Schoonbaert et al., 2007) or effects that are not significant at all (Shin, 2010). Under this account, language models' dependence on target language training and humans' dependence on prime language experience for structural priming would suggest that there are key differences between the models and humans in how grammatical representations function in comprehension and production.

An alternative reason for the absence of crosslingual structural priming effects for the English→Polish and English→Greek stimuli is a combination of model features and features of the languages themselves. For example, structural priming

effects at the syntactic level may overall be stronger for English targets. English is a language with relatively fixed word order, and thus, competence in English may require a more explicit representation of word order than other languages. In contrast to English, Polish and Greek are morphologically rich languages, where important information is conveyed through morphology (e.g. word inflections), and word orders are less fixed (Tzanidaki, 1995; Siewierska, 1993). Thus, structural priming effects with Polish and Greek targets would manifest as differences in target sentence morphology. However, contemporary language models such as XGLM have a limited ability to deal with morphology. Most state-of-the-art models use WordPiece (Wu et al., 2016) or SentencePiece (Kudo and Richardson, 2018) tokenizers, but other approaches may be necessary for morphologically rich languages (Klein and Tsarfaty, 2020; Park et al., 2021a; Soulos et al., 2021; Nzeyimana and Niyongabo Rubungo, 2022; Seker et al., 2022).

Thus, while humans are able to exhibit crosslingual structural priming effects between languages when the equivalent structures do not share the same word orders (Muylle et al., 2020; Ziegler et al., 2019; Hsieh, 2017; Chen et al., 2013), this may not hold for contemporary language models. Specifically, given the aforementioned limitations of contemporary language models, it would be unsurprising that structural priming effects are weaker for morphologically-rich target languages with relatively free word order such as Polish and Greek.

5.5.3 Implications for Multilingual Models

The results reported here seem to bode well for the crosslingual capacities of multilingual language models. They indicate shared representations of grammatical structure across languages (in line with Chi et al., 2020; Chang et al., 2022; de Varda and Marelli, 2023), and they show that these representations have a causal role in language generation. The results also demonstrate that crosslinguistic transfer can take place at the level of grammatical structures, not just specific phrases, concepts, and individual examples. Crosslinguistic generalizations can extend at least to grammatical abstractions, and thus learning a grammatical structure in one language may aid in the acquisition of its homologue in a second language.

How do language models acquire these abstractions? As Contreras Kallens et al. (2023) point out, language models learn grammatical knowledge through exposure. To the degree that similar outcomes for models and humans indicate shared mechanisms, this serves to reinforce claims of usage-based (i.e. functional) accounts of language acquisition (Tomasello, 2003; Goldberg, 2006; Bybee, 2010), which argue that statistical, bottom-up learning may be sufficient to account for abstract grammatical knowledge. Specifically, the results of our study demonstrate the in-principle viability of learning the kinds of linguistic structures that are sensitive to structural priming using the statistics of language alone. Indeed, under certain accounts of language (e.g. Branigan and Pickering, 2017), it is precisely the kinds of grammatical structures that can be primed that *are* the abstract linguistic representations that we learn when we acquire language. Our results are thus in line with Contreras Kallens et al.'s (2023) argument that it may be possible to use language models as tests for

necessity in theories of grammar learning. Taking this further, future work might use different kinds of language models to test what types of priors or biases, if any, are required for any learner to acquire abstract linguistic knowledge.

In practical terms, the structural priming paradigm is an innovative way to probe whether a language model has formed an abstract representation of a given structure (Sinclair et al., 2022), both within and across languages. By testing whether a structure primes a homologous structure in another language, we can assess whether the model's representation for that structure is abstract enough to generalize beyond individual sentences and has a functional role in text generation. As language models are increasingly used in text generation scenarios (Lin et al., 2022) rather than fine-tuning representations (Conneau et al., 2020a), understanding the effects of such representations on text generation is increasingly important. Previous work has compared language models to human studies of language comprehension (e.g. Oh and Schuler, 2023; Michaelov et al., 2022; Wilcox et al., 2021; Hollenstein et al., 2021; Kuribayashi et al., 2021; Goodkind and Bicknell, 2018), and while the degree to which the the mechanisms involved in comprehension and production differ in humans is a matter of current debate (Pickering and Garrod, 2007, 2013; Hendriks, 2014; Meyer et al., 2016; Martin et al., 2018), our results show that human studies of language production can also be reproduced in language models used for text generation.

5.5.4 Limitations

To ensure that the stimuli used for the language models indeed elicit structural priming effects in people, we only use stimuli made available by the authors

of previously-published studies on structural priming in humans. Thus, our study analyzes only a subset of possible grammatical alternations and languages. All of our crosslingual structural priming stimuli involve English as one of the languages, and all other languages included are, with the exception of Mandarin, Indo-European languages spoken in Europe. All are also moderately or highly-resourced in the NLP literature (Joshi et al., 2020). Thus, our study is not able to account for the full diversity of human language.

Additionally, while psycholinguistic studies often take crosslingual structural priming to indicate shared representations, there are alternate interpretations. Most notably, because structurally similar sentences are more likely to occur in succession than chance, it is possible that increased probability for same-structure target sentences reflects likely co-occurrence of distinct, associated representations, rather than a single, common, abstract representation (Ahn and Ferreira, 2023). While this is a much more viable explanation for monolingual than crosslingual priming, the presence of even limited code-switching in training data could in principle lead to similar effects across languages.

5.6 Conclusion

Using structural priming, we measure changes in probability for target sentences that do or do not share structure with a prime sentence. Analogously to humans, models predict that a similar target structure is generally more likely than a different one, whether within or across languages. We observe several exceptions,

which may reveal features of the languages in question, limitations of the models themselves, or interactions between the two. Based on our results, we argue that multilingual autoregressive Transformer language models display evidence of abstract grammatical knowledge both within and across languages. Our results provide evidence that these shared representations are not only latent in multilingual models' representation spaces, but also causally impact their outputs.

Acknowledgments

This chapter is a reprint of a paper published in the Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP).

Chapter 6

Crosslingual Structural Priming and the Pre-Training Dynamics of Bilingual Language Models

6.1 Introduction

This chapter follows up on the results of Chapter 5. In this chapter, we train bilingual models in order to remove the confounds encountered above with respect to training data quantities. By training our own models, we are also able to save the model at various points during training in order to understand when and how the shared abstract grammatical representations come about.

In contrast with the previous chapter, this chapter it not published or submitted for publication. It is a set of experiments, which explores questions that arose

from the previous chapters and related topics.

In this Section 6.3, I replicate the structural priming effects in Chapter 5. In Section 6.4, I demonstrate the time course of the emergence of structural priming effects over the course of training. In Section 6.5, I use a probing method to identify the shared representations that drive structural priming effects. In Section 6.6, I use a causal intervention method to try to establish a causal relationship between the representations I identified with a probe; however, there was no causal relationship. In Section 6.7, I test several potential causes for the lack of causal relationship between the probing results and the causal analysis.

6.2 Training Bilingual Language Models

We pre-train bilingual language models[1] to simulate the language experience of the bilingual participants in the human structural priming experiments we use here. We have two bilingual conditions. In the **simultaneous bilingual** setting, the models are exposed to one language (L1) during the first half of training and in the second half they are exposed to a mix of half L1 and half second language (L2) data. During the second half of training, batches were constructed with alternating sequences of L1 and L2 text, therefore at any given checkpoint it is possible to count precisely the number of tokens from each language that the model has seen. In the **sequential bilingual** setting, models are exposed only to L1 during the first half of training and then only to L2 in the second half of training.

We manipulate three factors: language pair (English-Dutch, English-Spanish,

[1] `https://huggingface.co/collections/catherinearnett/b-gpt-66f4b80e8fa8e95491948556`

English-Polish, English-Greek), language order (e.g. English L1, Dutch L2 and Dutch L1, English L2), and bilingual condition (simultaneous or sequential). As a result, we train a total of 16 language models. For example, for Dutch we train four models: Dutch-English simultaneous, Dutch-English sequential, English-Dutch simultaneous, and English-Dutch sequential.

We train a separate SentencePiece tokenizer (Kudo and Richardson, 2018) for each model. The tokenizer has the same language proportions as the model training data. For the simultaneous bilingual condition, the overall proportion of training data the model sees is 75% L1 and 25% L2 data. For the sequential bilingual condition, the overall proportion is 50% L1 and 50% L2 data.

The models are autoregressive GPT-2 Transformer language models with 124M parameters (Radford et al., 2018, 2019). Following Chang et al. (2023b), for each language, we take the first 128M lines of the deduplicated OSCAR corpus (Abadji et al., 2021). We tokenize the dataset with the tokenizer for each model. We create sequences of 128 tokens, shuffle sequences, and sample 2B tokens for the training set. We also create an evaluation set of 8k sequences (1M tokens). In total, the model trained for 128,000 steps. Starting at step 64,000, the model was then trained on either a mix of L1 and L2 (simultaneous) or just L2 data (sequential). We save the model at regular checkpoints over the course of training, and increase the number of checkpoints just after the introduction of L2 halfway through training. All model training details are reported in D.2.

6.2.1 Model Training

We report the mean surprisal[2] at each checkpoint for both languages each model is trained on. We calculate loss using the PsychFormers package[3] (Michaelov and Bergen, 2022c) to calculate the mean surprisal over the held out evaluation data. Figure 6.1 shows the loss curves for the English-Dutch models.

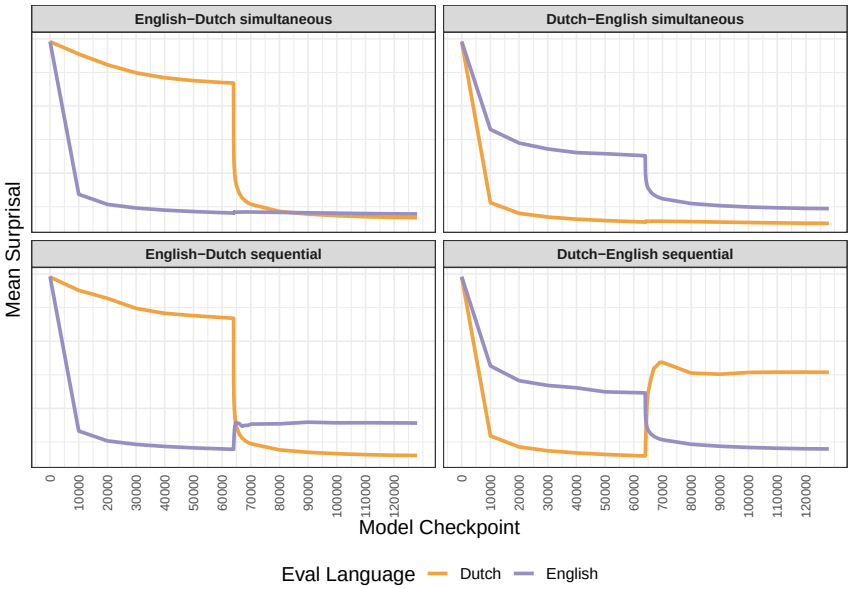

Figure 6.1: Mean surprisal is plotted for all checkpoints for the English-Dutch simultaneous (top left), Dutch-English simultaneous (top right), English-Dutch sequential (bottom left) and Dutch-English sequential (bottom right) models. Mean surprisal is plotted for both L1 and L2. Orange always indicates Dutch and purple always indicates English, irrespective of their L1/L2 status.

For both simultaneous bilingual models, we observe similar patterns: L1 mean

[2]Mean surprisal is equivalent to cross-entropy loss as defined in 1.1, normalized by the number of tokens. It is a measure closely related to perplexity, where perplexity is exponentiation of mean surprisal.

[3]Available at `https://github.com/jmichaelov/PsychFormers`

surprisal goes down quickly in the first half of training, while L2 mean surprisal stays relatively high. After the introduction of L2 at the halfway point, L2 loss drops quickly. Loss for both languages continues to slowly fall for the rest of training. There is an asymmetry between the English-Dutch and Dutch-English simultaneous models, where the English L2 loss drops much more quickly in the first half of training than does the loss for Dutch as L2. When Dutch is the L1 (top right, Fig. 6.1), the model is supposedly not being trained on English. We hypothesize that this is due to English contamination in the Dutch data. The reason we see an asymmetry is likely because there is not as as much Dutch contamination in the English data. This could be due to language use as many Dutch people speak English, but proportionally not as many English speakers also speak Dutch. It could also be due to differences in accuracy of language identification (LID) methods for English and Dutch, as English and Dutch are very similar languages. The loss curves are very different for the sequential condition models in the second half of training. After the model switches from being trained on L1 to L2 data, we see a sharp rise in the loss for the L1. The loss stays high for the rest of training. This is consistent with catastrophic forgetting (McCloskey and Cohen, 1989). This reflects the drastic shift in the distribution of training text from L1 to L2. As in the simultaneous condition, there is an asymmetry between the language order conditions. When English is the L1, there is a much less extreme rise in the L1 mean surprisal in the second half of training. This is also likely due to contamination. If the Dutch dataset contains English data, then the model is being exposed to small amounts of English data, which could be reducing the amount of catastrophic forgetting.

Comparing the English-Dutch loss curves with those of the other language pairs (Fig. 6.2), we see similar patterns. For the simultaneous models, especially when English is the L2, there seems to be a language similarity effect. Comparing the models in the second column from the left in Fig. 6.2, by the end of training, there is a much smaller difference between mean surprisal for English and Dutch and English and Spanish, relative to the differences in mean surprisal between English and Polish and English and Greek. The lower the mean surprisal for English, the greater the transfer benefit is from the L1. In the case of Dutch, which is the most similar to English of the four languages, the English performance benefits the most. Whereas for the Greek-English model, which is typologically and orthographically distinct from English, the English performance gets less of a boost. This is consistent with other work, which shows that linguistic similarity is one of the best predictors of successful crosslingual transfer (Chang et al., 2023a).

In the sequential condition, especially when English is the L1 (Fig. 6.2, second column from the right), there are differences in the magnitude of the catastrophic forgetting effect. For Dutch the increase in English mean surprisal is less than the increase for the Spanish and Polish, which in turn is less than that for Greek. This is also likely due to differences in linguistic similarity.

6.3 Structural Priming Effects

We detect structural priming effects by comparing the relative likelihood of a target sentence after different prime sentences. e

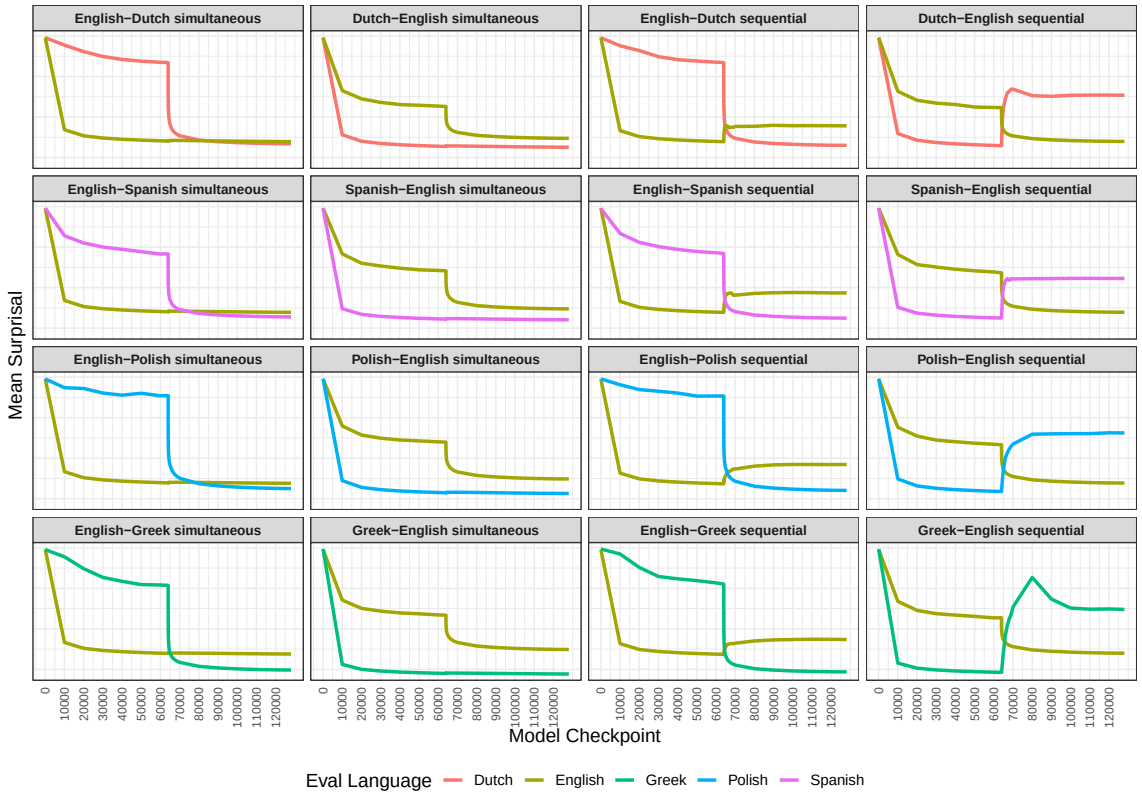

Figure 6.2: L1 and L2 mean surprisal for all models and all checkpoints. The color of each line indicates the evaluation language. Each facet represents one model.

We report crosslingual structural priming results, unless otherwise stated. This means that the prime sentence is in one language and the target language is in another language. In all of our language pairs, we compare comparable grammatical constructions, which both languages in each language pair share.

Comparing the probability of a prime sentence given two different contexts, structural priming demonstrates causal effects of shared abstract grammatical representations on model outputs without relying on access to internal model states. If a sentence with one grammatical construction in one language makes a sentence

with the same grammatical construction in another language more likely, then both sentences must be represented in the language model – at least in part – with a shared representation.

6.3.1 Calculating Structural Priming Effects

In human studies, structural priming effects are computed as the difference in normalized probability of a target sentence following each prime. Following Arnett et al. (2023), we measure the effect in language models by first calculating the surprisal of a target sentence given a prime sentence using the PsychFormers package (Michaelov and Bergen, 2022c). We then compute the normalized probability of each target sentence following each prime. For example, we compute the normalized probability P_N of a PO target T_{PO} following a PO prime P_{PO} as shown below, where T_{DO} is the DO target and P_{DO} would be a DO prime; see Eq. 6.1.

$$P_N(T_{PO}|P_{PO}) = \frac{P(T_{PO}|P_{PO})}{P(T_{PO}|P_{PO}) + P(T_{DO}|P_{PO})} \tag{6.1}$$

To test for a structural priming effect, we compare $P_N(T_{PO}|P_{PO})$ and $P_N(T_{PO}|P_{DO})$. If the former is significantly higher, this would indicate structural priming, because PO targets are more likely after PO primes than after DO primes. For each model and language combination, we fit a linear mixed effects model predicting the normalized probability of the target with prime type as a fixed effect and experimental item as a random intercept. Here, we only report results for the final model checkpoint, but we conduct the same tests for each model checkpoint. We report the results for the other checkpoints in Section 6.4 below. After fitting each linear mixed effect

model, we do a correction for multiple comparisons by controlling for false discovery rate (Benjamini and Hochberg, 1995).

6.3.2 Experimental Materials

We use the experimental stimuli from five studies across the four language pairs, covering three grammatical alternations: DO/PO, s-genitive/of-genitive, and Active/Passive.

DO/PO We use the Dutch and English stimuli from Schoonbaert et al. (2007), which contain pairs that contrast the Prepositional Object (PO) and Double Object (DO) dative constructions.

In some languages, for ditransitive sentences, when there are two objects, there are two possible ways to express the same event. One of these is the **Prepositional Object (PO)** construction (see example (1-a)). In this construction, the direct object 'hat' directly follows the verb and the indirect object is introduced with a prepositional phrase 'to the boxer'. The other is the **Double Object (DO)** construction (1-b). In this construction, the indirect object 'boxer' follows the verb, followed immediately by the direct object 'hat'.

(1) a. The cook shows a hat to the boxer. (PO)

b. The cook shows the boxer a hat. (DO)

(Schoonbaert et al., 2007)

Dutch has an equivalent alternation, with the same word order as English for PO

(Ex. (2-a)) and DO (Ex. (2-b)) sentences

(2) a. De kok toont een hoed aan de bokser.
The cook shows a hat to the boxer.
b. De kok toont de bokser een hoed.
The cook shows the boxer a hat.
(Schoonbaert et al., 2007)

s-genitive/of-genitive We use the Dutch and English stimuli from Bernolet et al. (2013), which contrast the two genitive constructions, which are semantically equivalent ways to express possession. In English, one of these is the **s-genitive** construction (Ex. (3-a)), where the possessor 'nun' is marked with ''s'. In this construction, the possessor 'nun' precedes the possessed thing 'egg'. In the **of-genitive** construction (Ex. (3-b)), the order is reversed and the possessed thing precedes the possessor. In this case, the preposition 'of' is used to express the possessive relationship.

(3) a. The nun's egg is yellow. (s-gen)
b. The egg of the nun is yellow. (of-gen)
(Bernolet et al., 2013)

Dutch has a similar alternation. For proper names, s-genitive possession can be marked with ''s', but for common nouns, possession is marked with the possessive pronoun that corresponds in gender to the possessor noun. In the example below (Ex. (4-a)), *non* 'nun' is feminine, so *haar* 'her' marks possession. Masculine nouns use *zijn* 'his' (Bernolet et al., 2013). The dutch of-genitive construction is more

similar to English, where the preposition *van* 'of' is used to show possession, and the order of the possessor and possessee is flipped, relative to the s-genitive order.

(4) a. De non haar ei is geel.
The nun POSS egg is yellow.
b. Het ei van de non is geel.
The egg of the nun is yellow.
(Bernolet et al., 2013)

Active/Passive For Spanish-English, Polish-English, and Greek-English experiments, we use stimuli that contrast active and passive constructions. For Spanish-English, we use stimuli from (Hartsuiker et al., 2004); for Greek-English, the stimuli come from (Kotzochampou and Chondrogianni, 2022); and for Polish-English, we use stimuli from (Fleischer et al., 2012).

Many languages allow events to be expressed as either active or passive. In **active** sentences, e.g. Ex. (5-a), the agent, or do-er of the action, 'the taxi' is the syntactic subject of the sentence, which in English, is marked by being the first argument in the sentence. The theme or patient, i.e. the thing having an action done to it, 'truck' is the syntactic object of the sentence and follows the noun. In **passive** sentences, the syntactic subject of the sentence is the theme. The agent is introduced in a prepositional phrase, 'by the taxi' (Ex. (5-b)).

(5) a. The taxi chases the truck. (Active)
b. The truck is chased by the taxi. (Passive)
(Hartsuiker et al., 2004)

Spanish expresses active and passive sentences very similar to English, following the same word order (Ex. (6-a) and (6-b), respectively).

(6) a. El taxi persigue el camión.
The taxi chases the truck.
b. El camión es perseguido por el taxi.
The truck is chased by the taxi.
(Hartsuiker et al., 2004)

Typologically, Polish and Greek are more different from English than either Dutch or Spanish is. Both of these languages mark the syntactic subjects and objects using case marking, unlike English, Dutch, and Spanish, which do this only with word order. In Polish, for example, in the active, *sportowiec* 'sportsman' is in the nominative case and is the syntactic subject of the sentence. The patient 'ballet dancer' takes the accusative and is the grammatical object of the sentence. In the passive, it is in the accusative case (*sportowca*) and is introduced with a prepositional phrase. The patient 'ballet dancer'), in this case, is in the nominative case.

(7) a. Sportowiec przygniata baletnicę.
sportsman.NOM.SG squash.PRES.3SG ballet-dancer.ACC.SG
"The sportsman squashes the ballet dancer."

b. Baletnica jest przygniatana przez
ballet-dancer.NOM.SG be.3SG.PRES squash.PST.PART by
sportowca.
sportsman.ACC.SG
"The ballet dancer is squashed by the sportsman."

(Fleischer et al., 2012)

Similarly, Greek marks subject and object roles with case marking. When it is

the subject, αθλητής (*athlitis*) 'athlete' is nominative, but as an object, it takes the accusative case (αθλητή, *athliti*). Greek, unlike Polish or the other languages described here, has a specific verbal morphology to encode active or passive voice (cf. (8-a) and (8-b)), therefore the verb form is also specific to passive voice, unlike the other languages shown here, which use a combination of the present copula and the past participle to mark passive voice.

(8) a. Ο αθλητής κλωτσ**άει** τον κλέφτη.
O athlitis klots**aei** ton klefti.
The athlete.NOM kicks-ACTIVE the thief.ACC.

"The athlete kicks the thief."

b. Ο κλέφτης κλωτσι**έται** από τον αθλητή.
O kleftis klots**iete** apo ton athliti.
The thief.NOM kicks-**PASSIVE** by the athlete.ACC.

"The thief is kicked by the athlete."

(Kotzochampou and Chondrogianni, 2022)

The Spanish, Greek, and Polish experiments have many fewer stimuli pairs than the Dutch experiments. Because we do not primarily aim to replicate human experimental results, we create new prime-target pairs by combining sentences into every possible order. Then we randomly sample pairs so that we have 144 sets for each the Spanish, Greek, and Polish stimuli, otherwise the experiment is over-powered. For each set, we have 16 possible conditions, for a total of 2304 observations per stimuli set. This matches the amount of statistical power for the Dutch experimental materials.

6.3.3 Results

Overall, we replicate the crosslinguistic structural priming effects[4] in Arnett et al. (2023) (Fig. 6.3). In all cases, when English is the target language, we find that a target is more likely if the prime sentence matches its grammatical structure. We also find that for the experiments with Schoonbaert et al. (2007) and Kotzochampou and Chondrogianni (2022) stimuli, when English is the prime language, we also find structural priming effects. There is still a numerical effect in the expected direction for the experiments with Bernolet et al. (2013) and Hartsuiker et al. (2004) stimuli where English is the prime language.

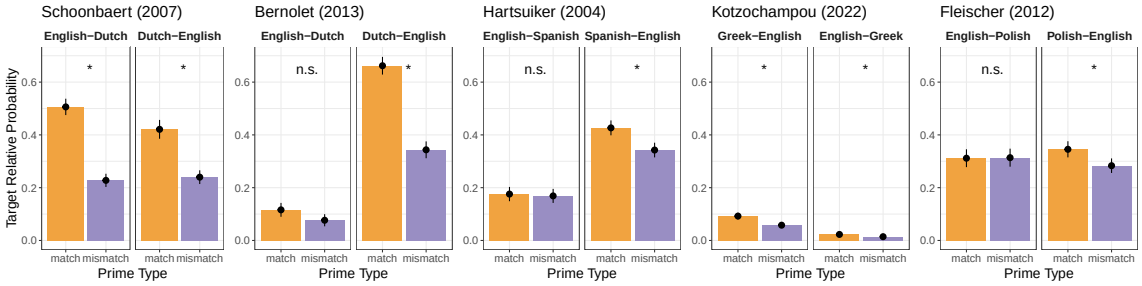

Figure 6.3: Priming results for the simultaneous bilingual conditions. For all experiments, prime language corresponds to L1 and target language corresponds to L1. Significance is indicated with *. Color indicates prime condition. Orange indicates congruent or matching prime and target types and purple indicates mismatched prime and target types.

Arnett et al. (2023) failed to demonstrate robust structural priming effects in language models for English-Polish and English-Greek. In that paper, we hy-

[4]Following results from the human structural priming literature, where it has been found that structural priming effects are strongest when the prime language is the participant's L1, and the target language is the L2, we only report results from the L1→L2 priming conditions. We report L2-L1 priming results in D.3.

pothesized that this was due to the much lower amount of training data that the models had seen from these languages. After controlling for this by training custom language models, we find significant structural priming effects for these languages in the simultaneous condition.

There is an asymmetry in the results, where we see more robust structural priming effects when English is the target language, as opposed to when English is the prime language. One possible explanation is that English is more sensitive to word order than other languages. We further discuss this in Section 6.3.4. This has not been observed in the human psycholinguistics literature, because English is almost always the L2 for human experiments (and thus the target language). The only exception from this set of experiments is Schoonbaert et al. (2007). This is due to the populations which are usually sampled from for these experiments, namely university students in countries like the Netherlands, where it is easiest to find L1 Dutch and L2 English speakers, as opposed to L1 English and L2 Dutch speakers. This is a major confound in the literature. Therefore, this asymmetry has never been observed in human experiments. Using language models to develop and test these hypotheses can help overcome some of the experimental confounds that are difficult to control for in human experiments. This approach, called *in silico* experimentation, has also been proposed in neuroscience research, where traditional experiments are costly and time-intensive (Jain et al., 2024).

Another possible explanation for this asymmetry is data contamination. We hypothesize that it is more likely that English data contaminates datasets for other languages than vice versa. Therefore, in the cases where English is the target lan-

guage (and therefore the L2), English has actually been seen more than intended because of contamination. This could be boosting the structural priming effects, especially when English is the target language.

We find similar patterns from the sequential bilingual models (Fig. 6.4). Despite evidence that the sequential models experienced catastrophic forgetting of the L1 (Section 6.2.1), the Dutch and Spanish models still exhibit structural priming effects in the final checkpoints, and we see significant structural priming in the English-Polish condition. However, there is a reduced effect size, which is likely caused by the catastrophic forgetting, which may cause the models to represent knowledge of L1 less well by the end of training. There are stronger effects for Dutch and Spanish, and less strong effects for Greek and Polish. This is likely another language similarity effect.

There are several key ways in which these languages differ, including writing system, case morphology, and how the grammatical alternations are encoded. For instance, English, Dutch, Spanish, and Polish all use periphrastic constructions to encode the passive voice, whereas Greek uses verbal morphology to do so. Greek is also the only language that uses a non-Latin writing system. Therefore, we are not able to disentangle these factors. We can only hypothesize that whether these languages share these features has an impact on crosslingual transfer.

6.3.4 Word Order Analysis

As mentioned above, one explanation for the stronger structural priming effects is the increased importance of word order in English relative to other languages

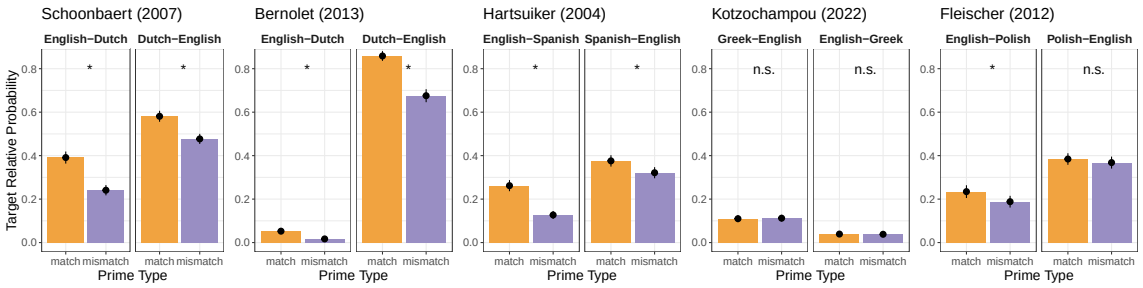

Figure 6.4: Priming results for the sequential bilingual conditions. For all experiments, prime language corresponds to L1 and target language corresponds to L1. Significance is indicated with *. Color indicates prime condition. Orange indicates congruent or matching prime and target types and purple indicates mismatched prime and target types.

(Arnett et al., 2023). We see much weaker structural priming effects when either Polish or Greek are prime or target languages. Both of these languages have more flexible word order because of overt morphological marking like case marking (Siewierska, 1993; Tzanidaki, 1995). Because of a syntactic feature, we can begin to test whether the weaker structural priming effects in Polish are due to more flexible word order.

In addition to an Active-Passive alternation, Polish has a third alternative: OVS (Fleischer et al., 2012). The default word order for Polish is SVO. In active sentences (Ex. (9-a)), the order of the arguments is the syntactic subject and agent 'sportsman'), followed by the verb, followed by the syntactic object and theme 'ballet dancer'). In passive sentences (Ex. (9-b)), the order of the argument is first and the syntactic subject, marked by the nominative case, is the theme. The syntactic object, marked by accusative case and part of a prepositional phrase, is the agent. In OVS sentences, the grammatical construction is active, as the syntactic subject is the agent and then syntactic object is the theme; however, the order of the arguments is

the same as the passive (ballet dancer, then verb, then sportsman; Ex. (9-c)).

(9) a. Sportowiec przygniata baletnicę.
sportsman.NOM.SG squash.PRES.3SG ballet-dancer.ACC.SG
"The sportsman squashes the ballet dancer."

b. Baletnica jest przygniatana przez sportowca.
ballet-dancer.NOM.SG be.3SG.PRES squash.PST.PART by sportsman.ACC.SG
"The ballet dancer is squashed by the sportsman."

c. Baletnicę przygniata sportowiec.
ballet-dancer.ACC.SG squash.PRES.3SG sportsman.NOM.SG
"The sportsman squashes the ballet dancer."

If structural priming effects are driven by constructions alone, independent of word order, then we would expect to see that OVS primes Active sentences and vice versa. If structural priming effects can be explained by word order, then we would expect OVS to prime Passive and vice versa, as the two sentence types share the same argument order.

We created a new set of stimuli. For Polish-English priming, we use the same procedure, but add the OVS as a prime, so there are three total prime conditions: Active, OVS, and Prime. In English-Polish priming, there are still only two prime conditions (Active and Passive), as English does not have an OVS structure. We evaluated the structural priming effects of both Active and OVS as targets. We used the English-Polish and Polish-English simultaneous models and evaluated only on the final model checkpoint. We evaluate each model on both priming directions, i.e. on both L1-L2 and L2-L1 priming. In Figure 6.5, we show the results of Polish as

a prime language and English as a target language. There are three primes because Polish has all three constructions. By contrast, Figure 6.6 shows the results with English as a prime language and Polish as a target language. As a result, there are only two prime types.

In Polish-English priming, Active targets are less likely after a Passive prime than after either an Active or OVS prime. OVS primes Active to a similar degree as Active primes Active. Fleischer et al. (2012) only tested Polish-English priming, but we also test English-Polish priming. There is no human baseline to compare the results to, however, was test whether Active or Passive primes lead to higher OVS target probability. We find that Active primes OVS more than Passive does (Fig. 6.6. Therefore, in both priming language directions, Active primes OVS.

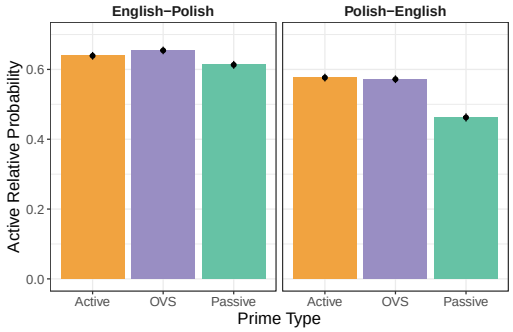

Figure 6.5: Polish-English priming.

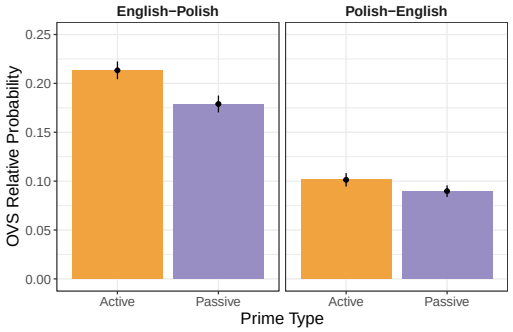

Figure 6.6: English-Polish priming.

Figure 6.7: Each facet represents a different simultaneous bilingual model accord to the order of language exposure, either English-Polish or Polish-English both model language orders.

These results diverge from the human experimental results in Fleischer et al. (2012), in which the authors found that OVS primed Passive sentences in Polish-English priming. The results support the former hypothesis, that structural priming

is about the construction itself, as opposed to word order, at least in language models. These results do not suggest word order is the reason for asymmetrical structural priming effects. It is possible the effect asymmetry is due to training data contamination, or another reason that has not yet been considered.

As the results diverge from the human experimental results, it seems the mechanism behind structural priming may be different for humans and language models. This does not necessarily mean that language models cannot be used for *in silico* experimentation, but it is important to recognize where the limitations are for language models as computational models of human language processing.

6.4 Training Dynamics of Structural Priming

Here we report the structural priming effects over the course of training, which were calculated in the same way as the previous section, but for each model checkpoint. For each checkpoint, we plot the relative probability of the target depending on the two primes. Prime condition is indicated by color, where congruent prime is orange, and incongruent prime is purple. The structural priming effect size is the magnitude of the difference between the two relative probabilities at a given checkpoint. Figure 6.8 shows the structural priming effects at each checkpoint. We report the results for all models, language directions, etc. in D.4.

Simultaneous bilingual condition Across all models, we find that structural priming effects do not emerge until after exposure to L2, even when we test the L1 as the target language. This means that structural priming shows evidence of

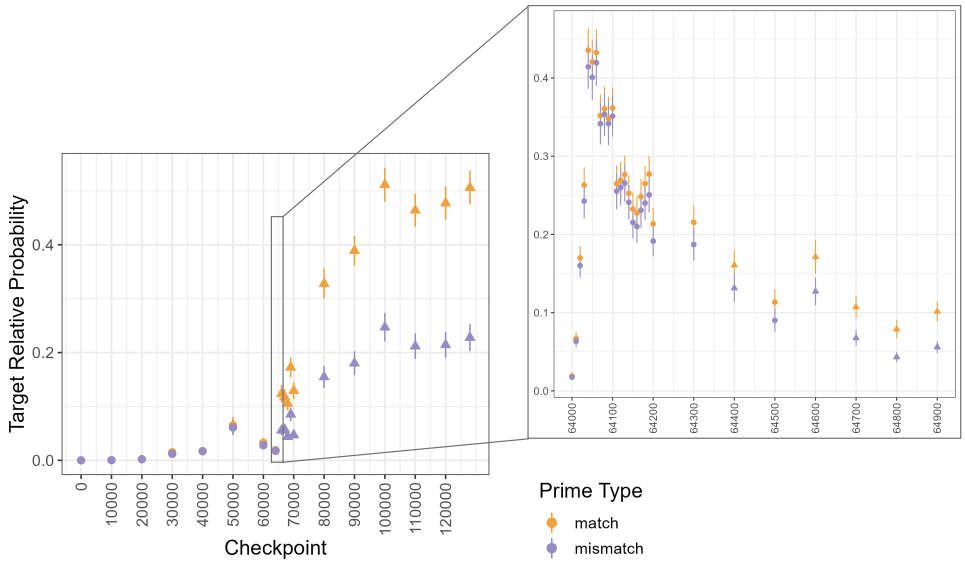

Figure 6.8: The figure on the left shows structural priming effects for English-Dutch priming for the simultaneous bilingual model, evaluated on Schoonbaert et al. (2007) stimuli. Significant structural priming effects are marked with triangles, effects that are not significant are marked with circles. In the figure on the right, we plot the structural priming effects for the first 900 steps after L2 exposure, where we saved more fine-grained checkpoints.

shared representations, not of zero-shot transfer, as we only observe structural priming effects once the models have updated their representations based on L2 training data. This also shows that these effects are not driven entirely by data contamination. While we hypothesize that some of the asymmetries in language direction are caused by contamination, the amount of contamination alone is not enough to lead to structural priming effects.

After exposure to L2, structural priming effects emerge quickly. In Figure 6.8, the effect emerges 600 steps[5] after first exposure to L2, at which point the lan-

[5]At earlier steps, there are significant structural priming effects, but they are not stable across

guage model has seen approximately 5M L2 tokens. For most models where we find structural priming effects, the effects emerge within the first 1000 steps after L2 exposure. These results contribute to our understanding of crosslingual transfer and multilingual representations. Ongoing research has demonstrated that contamination with other languages can impact multilingual model performance (Blevins and Zettlemoyer, 2022; Muennighoff et al., 2023). For example, Muennighoff et al. (2023) find zero-shot crosslingual transfer on XNLI for models that are not intentionally trained on some of the XNLI languages. However, in an analysis of the pre-training dataset, they find small amounts of data in non-included languages (e.g. approximately 0.006% of the data is in Thai, corresponding to roughly 20M tokens). So we cannot attribute these results to crosslingual transfer, but instead they are likely driven by contamination. The results presented here show that with even as few as 5M tokens – albeit in a much smaller model – can lead to behaviors we attribute to multilingual representations. This suggests a reconsideration of other "zero-shot" crosslingual capabilities in multilingual language models.

Sequential bilingual condition We see slightly different patterns for the sequential bilingual condition. In the simultaneous condition, it was more common for the structural priming effect size to continue to increase over the course of training (Fig. 6.9, left). In the sequential bilingual condition, where we observed increased mean surprisal for L1 consistent with catastrophic forgetting, we observe in some cases a larger structural priming effect size earlier on in the second half of training. Then, in the later stages of training, the effect size starts to shrink (Fig. 6.9, right).

checkpoints until this checkpoint.

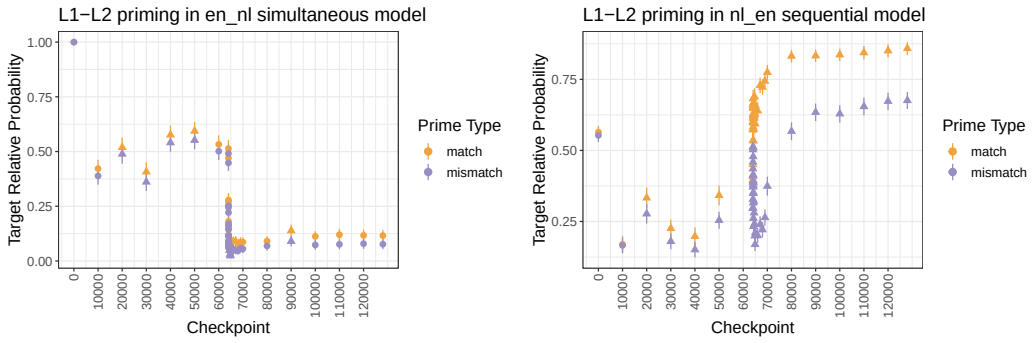

Figure 6.9: Structural priming effects over the course of training for Dutch-English models evaluated on Bernolet et al. (2013) stimuli, for the simultaneous (left) and sequential (right) bilingual conditions.

In other cases, we see that the structural priming effect is significant, but then disappears completely (Fig. 6.10). In the plots in the left column, which show structural priming effects over the whole course of training, that there are structural priming effects between step 64000 and approximately steps 70000 in the case of Spanish and 80000 in the case of Polish. The plots on the right show results for the same models, but only for the first 1000 steps after exposure to L2. The structural priming effects grow in this window, but ultimately disappear at a later stage in training.

6.4.1 Mean Surprisal and Structural Priming Effects

If structural priming is conditioned on the model having some knowledge of L2, as opposed to being driven by zero-shot crosslingual transfer, then we should expect the emergence of structural priming effects to be temporally linked to L2 proficiency. We measure L2 proficiency with two metrics: mean surprisal (Section

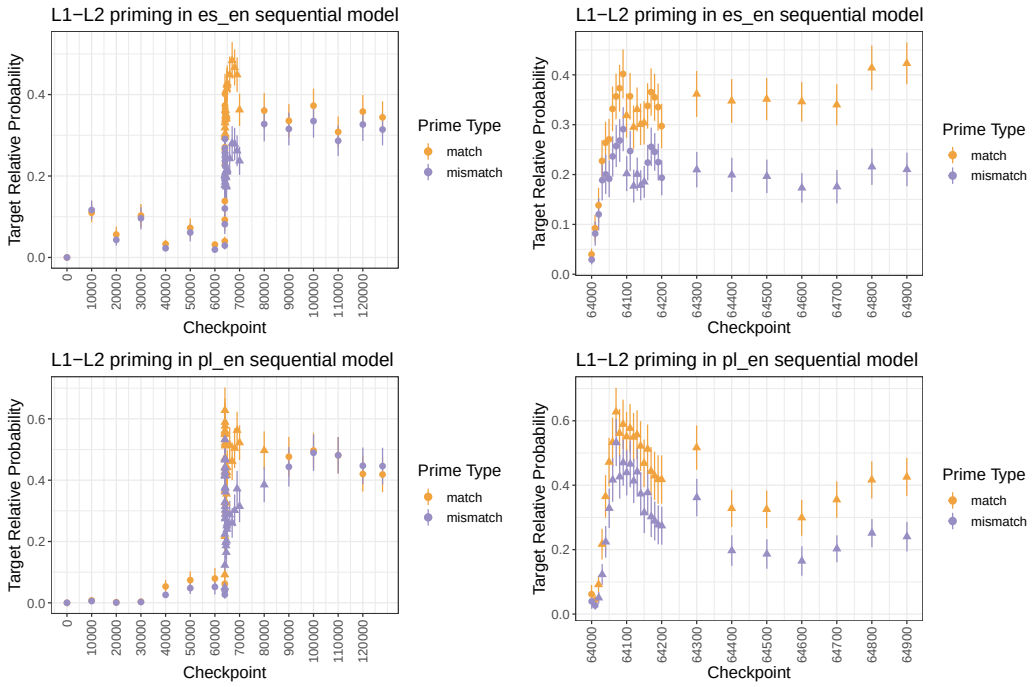

Figure 6.10: Structural priming effects over the course of training for Spanish-English (top) and Polish-English (bottom) sequential bilingual models evaluated on Hartsuiker et al. (2004) and Fleischer et al. (2012) stimuli, respectively. For each model, we show both the structural priming effects for all checkpoints (left) and the first 900 checkpoints after L2 exposure (right).

6.4.1) and BLiMP (Section 6.4.2).

Here we plot mean surprisal and structural priming effect size. In these plots, instead of plotting each prime condition separately, the black line represents the difference between the two conditions. Higher values represent larger effect sizes. In pink, we plot the mean surprisal. The axes scales are rescaled for easier comparison. In Figure 6.11 (top left, bottom left), we show two examples from the simultaneous bilingual condition. We plot the L2 mean surprisal. In the first half of training,

we see high L2 mean surprisal, which indicates that the model does not have very good representations of L2. Correspondingly, structural priming effects are very small or non-existent. At the halfway point of training, there is a sudden drop in the mean surprisal. At the same time, the structural priming effect size increases rapidly. This supports our claim that structural priming effects correspond with the models learning representations for both languages, as opposed to zero-shot using L1 representations to process L2 text.

In Figure 6.11 (top right and bottom right), we plot structural priming effect and mean surprisal for the sequential bilingual condition. In the second half of training, there is a rapid rise in L1 mean surprisal, consistent with catastrophic forgetting. During this period, structural priming effect size increases, rapidly. Then, as the L1 mean surprisal plateaus, structural priming effect sizes decrease and remain small (top right) or disappear altogether (bottom right). We argue that the structural priming effects are consistent with the model transferring shared abstract grammatical representations learned from L1 to represent L2. In the initial period, the representations of L1 and L2 are the most similar, which explains the larger structural priming effects. Then, as training continues, the representations drift such that they represent L2 much better than L1. As a result, the structural priming effect size shrinks.

6.4.2 BLiMP analysis

Some argue that measures like perplexity do not accurately measure language model performance (Kuribayashi et al., 2021), therefore we also evaluate each model

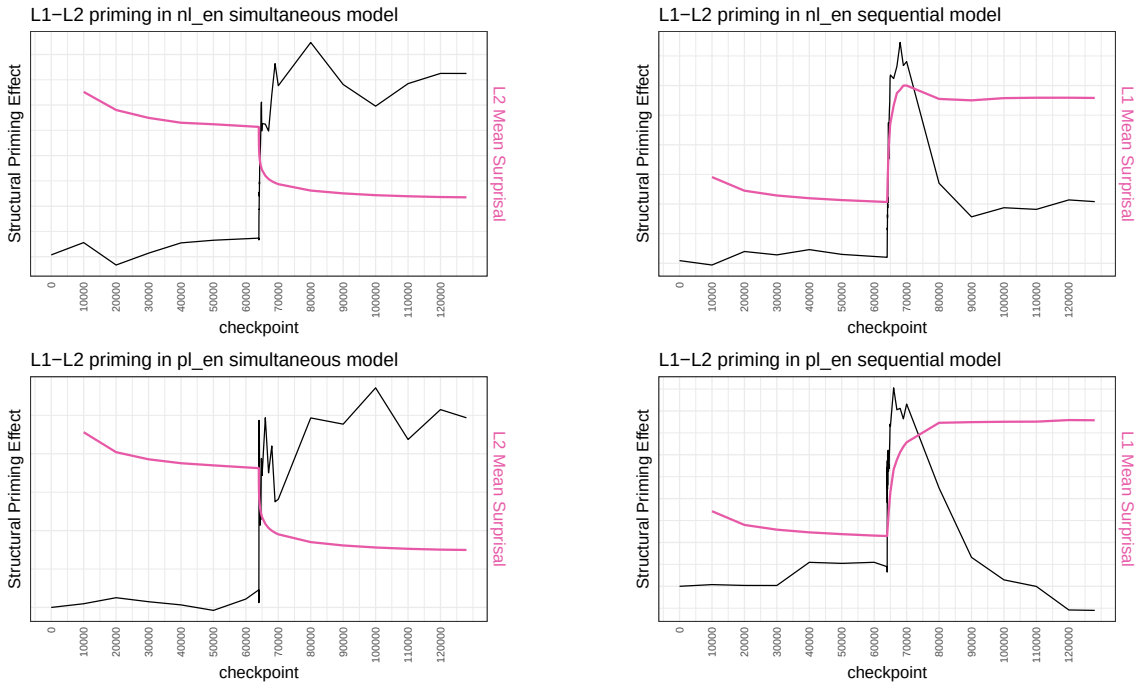

Figure 6.11: Structural priming effect (black) and L2 mean surprisal (pink) plotted over the course of training. Top left: Dutch-English simultaneous condition. Top right: Dutch-English sequential condition. Bottom left: Polish-English simultaneous condition. Bottom right: Polish-English sequential condition.

checkpoint on the BLiMP benchmark (Warstadt et al., 2020). BLiMP demonstrates the grammatical knowledge of the model, which is predictive of a model's ability to generate grammatical text. We evaluate each model checkpoint on BLiMP using the LM Evaluation Harness (Gao et al., 2024), and we report the average score over all sub-tasks. While there are BLiMP benchmarks for other languages, BLiMP does not exist for all other languages in our sample. Therefore, we limit our analysis to English BLiMP. We report results for all models in D.6.

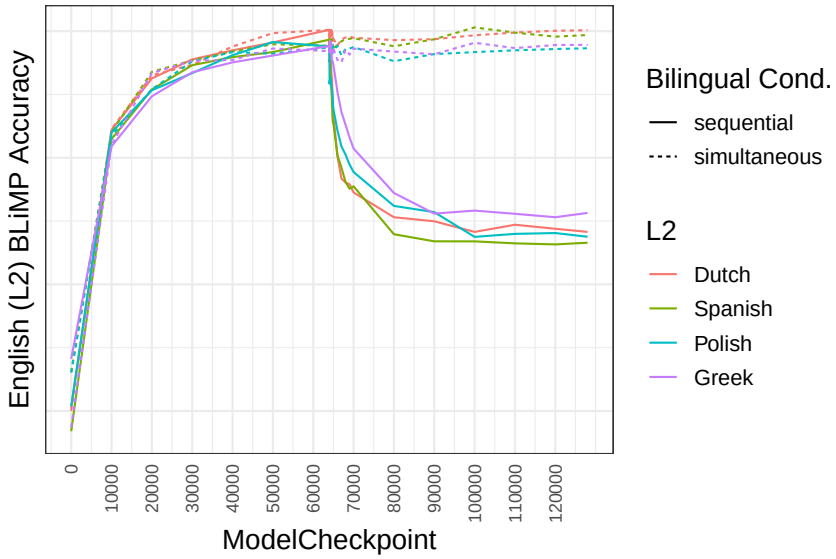

Figure 6.12: English L1 models in both the sequential (solid lines) and simultaneous (dotted lines) conditions. BLiMP accuracy is plotted over the course of training.

For models where English is the L1, we see differences in BLiMP scores over the course of training according to the bilingual conditions (Fig. 6.12. In the simultaneous bilingual condition, there is a small dip in BLiMP score after exposure to L2, but then the scores rise again and stay at ceiling. In the sequential bilingual condition, BLiMP scores fall rapidly after exposure to L2. At about step 80000, performance plateaus. The performance never returns to the level of the model at checkpoint 0, but BLiMP score at the final checkpoint is worse than at checkpoint 10000 for all models. This further supports the observation that the models in the sequential bilingual condition experience catastrophic forgetting. It is even more noteworthy, therefore that the models exhibit structural priming effects during the period where L1 mean surprisal rises and BLiMP scores fall.

Comparing BLiMP performance for the models in the simultaneous condition,

we observe a difference in final checkpoint performance. Dutch models have the best performance, followed by Spanish. Greek and Polish again show the worst performance. These results demonstrate differential crosslingual transfer benefits. The language that is the most similar to English (Dutch) leads to the highest BLiMP scores, followed by Spanish, which is also very similar to English. Polish and Greek are the most different from English and show the least benefit from crosslingual transfer. This is also consistent with previously demonstrated effects of linguistic similarity (Chang et al., 2023a).

Comparing BLiMP scores to structural priming effect size (Fig. 6.13), we see the same patterns as those from mean surprisal. Again, we plot both structural priming and BLiMP accuracy and scale the axes for easier comparison. In the simultaneous condition, when English is L2, structural priming effects only emerge when BLiMP scores rise suddenly. Structural priming effect size patterns with BLiMP accuracy. They both rise very quickly after exposure to L2 and then plateau.

In the sequential condition, BLiMP accuracies also show evidence of catastrophic forgetting (Fig. 6.14, right), such that BLiMP accuracy rises in the first half of training, but falls dramatically after the model stops being exposed to L1. BLiMP, therefore, reflects the same patterns we see in the perplexity results. However, as with the perplexity results, when BLiMP scores fall as a result of catastrophic forgetting, structural priming effects do not always necessarily disappear.

We plot the relationship between mean surprisal and BLiMP accuracy (Fig. 6.15. There is a clear relationship, thus we argue that mean surprisal is a good indicator of early model learning and of the model's ability to do next word prediction

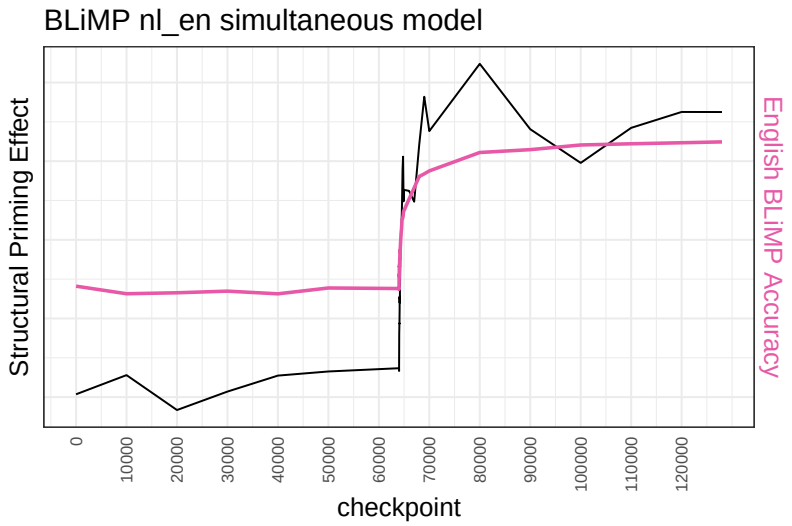

Figure 6.13: Dutch-English structural priming effects and English BLiMP accuracy plotted over the course of training.

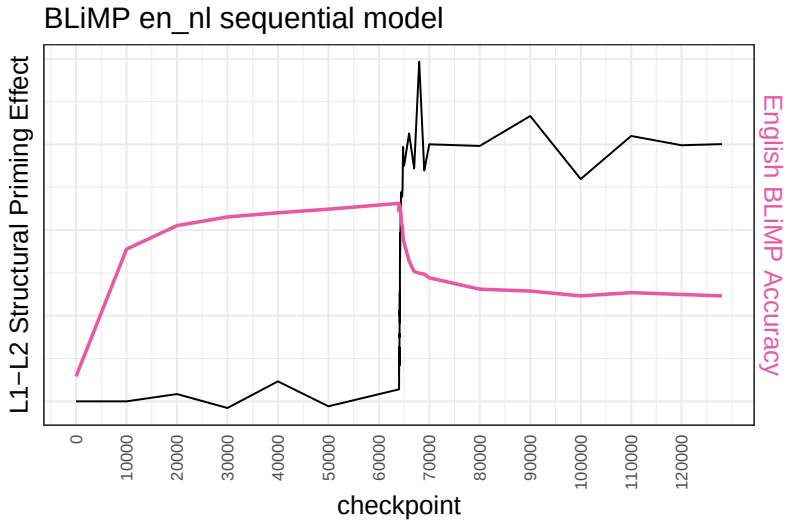

Figure 6.14: English-Dutch structural priming effects and English BLiMP accuracy plotted over the course of training.

and generate grammatical text. Some work has shown that perplexity is predictive of model performance (Xia et al., 2023), but it is not predictive of all metrics of

model performance (Kuribayashi et al., 2021), especially higher-order capabilities like reasoning. We argue for small models or for the early stages of model training (or in the case of wanting only to evaluate the model's next word prediction capabilities), perplexity is a good performance metric. This is important, as many languages – especially low-resource languages – do not have many, if any, benchmarks. Evaluating models trained on these languages is very difficult. But evaluating perplexity is possible as long as there is held-out text data for a language.

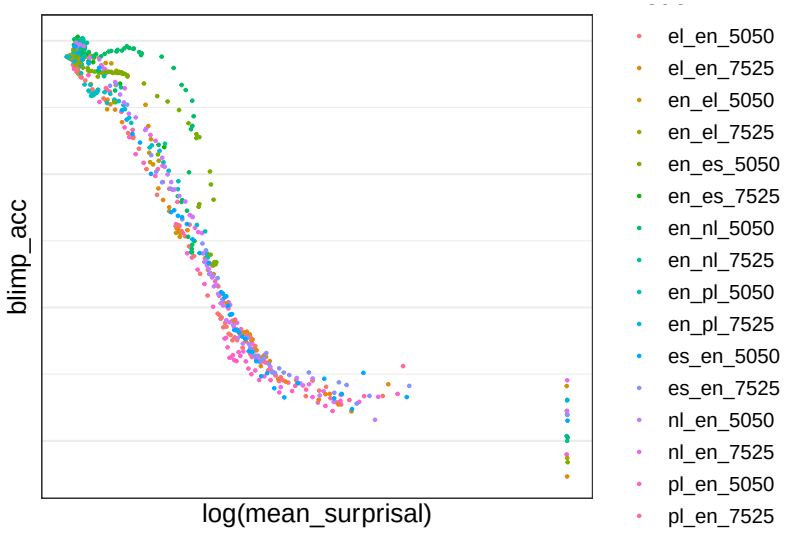

Figure 6.15: Relationship between log mean surprisal and BLiMP accuracy.

6.4.3 Interim Discussion

In this section, we have demonstrated when structural priming effects emerge over the course of model training. L2 performance, as measured by both mean surprisal and BLiMP accuracy, seems to be a necessary condition for structural priming effects. In all cases where models exhibit crosslingual structural priming effects, per-

formance for L2 is high. For L1, if the model exhibits catastrophic forgetting, this does not necessarily entail structural priming effects.

Now that the time course has been established, we ask about the nature of the shared representations underlying structural priming. Given that the model must have some knowledge of L2 in order to exhibit structural priming effects, is it the case that the models are developing new representations that can be used for both L1 and L2, or is the model adapting the existing L1 representation for L2? In the following section, we analyze the internal model activations to address this question.

6.5 Locating Shared Abstract Grammatical Representations

The linear representation hypothesis states that high-level concepts are represented linearly in the embedding space of language models (Park et al., 2024; *inter alia*). Several studies have shown that syntactic information is among the type of conceptual information that can be identified with linear probes (Lin et al., 2019; Hewitt and Manning, 2019; Arps et al., 2022). Therefore, we predict that the shared abstract grammatical representations driving structural priming effects in language models should be identifiable as linear representations.

Most of the mechanistic interpretability studies on grammatical structures have focused on grammatical relations, as defined by the Universal Dependency framework (Chi et al., 2020; Maudslay and Cotterell, 2021; Arps et al., 2022, *inter alia*). In this set of experiments, we seek to identify higher-order grammatical

representations, which involve multiple arguments and can comprise entire sentences.

To identify the representations, we save the intermediate model states at every layer. We use a linear classifier to find the linear axis along which the model hidden states can be linearly classified. This linear classifier is referred to as a *probe*.

We identify a linear probe at each layer. We do this because it has been shown that syntactic information is best identified in middle layers of the model (Peters et al., 2018; Tenney et al., 2019; Chi et al., 2020; Xu et al., 2022b; Weissweiler et al., 2023b). We expect that we will have the highest classification accuracy in the middle layers.

As the model processes the stimuli token-by-token, we create sentence embeddings for each layer of the model. We do this using SGPT (Muennighoff, 2022), which creates sentence embeddings, $\vec{s}$, by weighting the activations for each token according to its position. The first token has a weight of 1, the second token has a weight of 2, and so on. Therefore the activations for token at position i (with indexing starting at 1), will have a weight of i. The weighted activations are then summed to create the sentence embedding. See Equation 6.2.

$$\vec{s} = \sum_{i=1}^{n} i \cdot \vec{t_i} \tag{6.2}$$

We modified SGPT to generate weighted mean embeddings individually for each layer.

In order to create the training data for our linear probe, we take minimal pairs from the stimuli used in each of the experiments above. For example, a sentence which is identical except in whether it is a DO or PO double object sentence (Ex.

(1-a) and (1-b), reprinted below for ease of reading).

a. The cook shows a hat to the boxer. (PO)

b. The cook shows the boxer a hat. (DO)

We then fit our linear probe using Linear Discriminant Analysis (LDA), which identifies a linear classifier that best discriminates the activations from the two types of sentences. We fit separate LDA axes for each language and grammatical alternation. We then measure the classification accuracy of the probe.

In Figure 6.16, we plot the LDA score for each layer. In the early layers, there is low classification accuracy, but in middle and later layers, accuracy is much higher, sometimes reaching ceiling accuracy. In most cases, accuracy begins to rise from layer 3, and peaks between layers 7 and 10. In some cases, accuracy drops in the final layers. This is consistent with previous findings about the middle layers containing the most syntactic information. We report the layer-by-layer classification results for other datasets in Figure D.7.2.

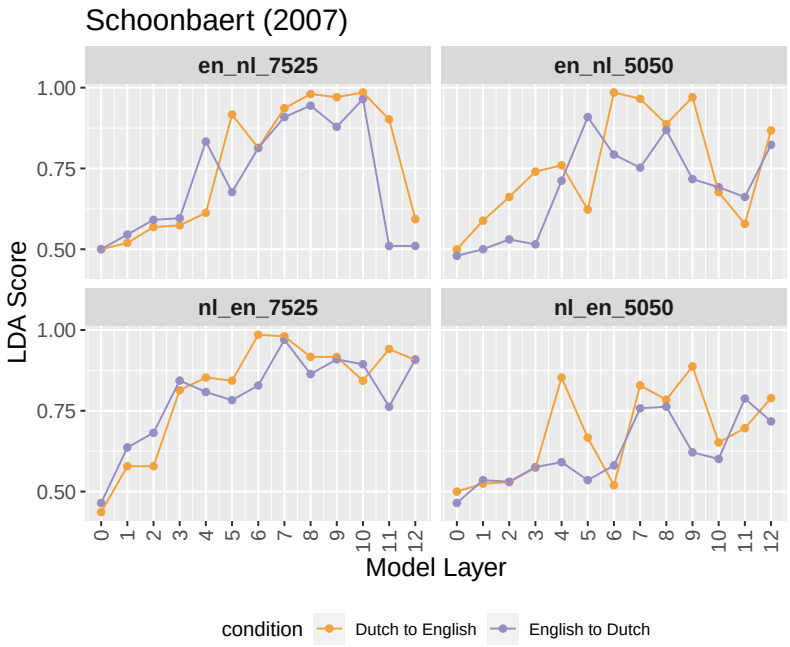

Figure 6.16: Classification accuracy for English-Dutch simultaneous (top left), English-Dutch sequential (top right), Dutch-English simultaneous (bottom left), and Dutch-English sequential (bottom right) models for both Dutch→English (orange) and English→Dutch priming (purple).

6.5.1 LDA and Training Dynamics

Next, we conduct the same analysis, but for every model checkpoint. We train a classifier on for every model checkpoint and evaluate each of the classifiers on every checkpoint. We analyze the results with the aim to understand, first, whether the representations emerge at the same point in training as when the structural priming effects emerge. If the representations are only identifiable at the checkpoints when we see structural priming effects, this helps confirm that the representations we identified are linked to the relevant grammatical structures.

Second, we evaluate whether the linear representations are stable across check-

points. Is it the case that the representations from L1 are mapped onto L2, or does the model develop new language-neutral representations?

Figure 6.17 shows the classification accuracy of the LDA fit to the activation patterns of each checkpoint and each layer for L1 at classifying the activation patterns for L2 at every other checkpoint. In these plots, each facet represents a different layer of the model. Across the different models, the highest classification accuracy is seen in the middle layers. Classification accuracy, as measured by LDA score, is represented by the color of each cell. Higher LDA scores are represented by blue cells, where orange cells are at or around chance at classification.

For the Dutch-English simultaneous model and the Schoonbaert et al. (2007) stimuli, classification accuracy is at about chance in the evaluation checkpoints in the first half of training. When evaluated on the checkpoints in the second half of training, the accuracy is high. In the middle and late-middle layers, accuracy is at or near ceiling in many cases.

The classifiers trained on the early checkpoints show low accuracy across all evaluation checkpoints, likely because the model has not learned any representations for L1 in the early checkpoints. Therefore, the activations do not show patterns according to grammatical structure. By checkpoint 30000, the classifier is able to achieve high accuracy. This suggests that it is within the first 10000 to 20000 steps that the representations of these grammatical constructions are learned for L1.

There is also high classification accuracy for linear probe trained on earlier checkpoints on the activations of later checkpoints. For example, in layers 7 through 10, the probe trained on the early-middle checkpoints for L1 achieves high accuracy

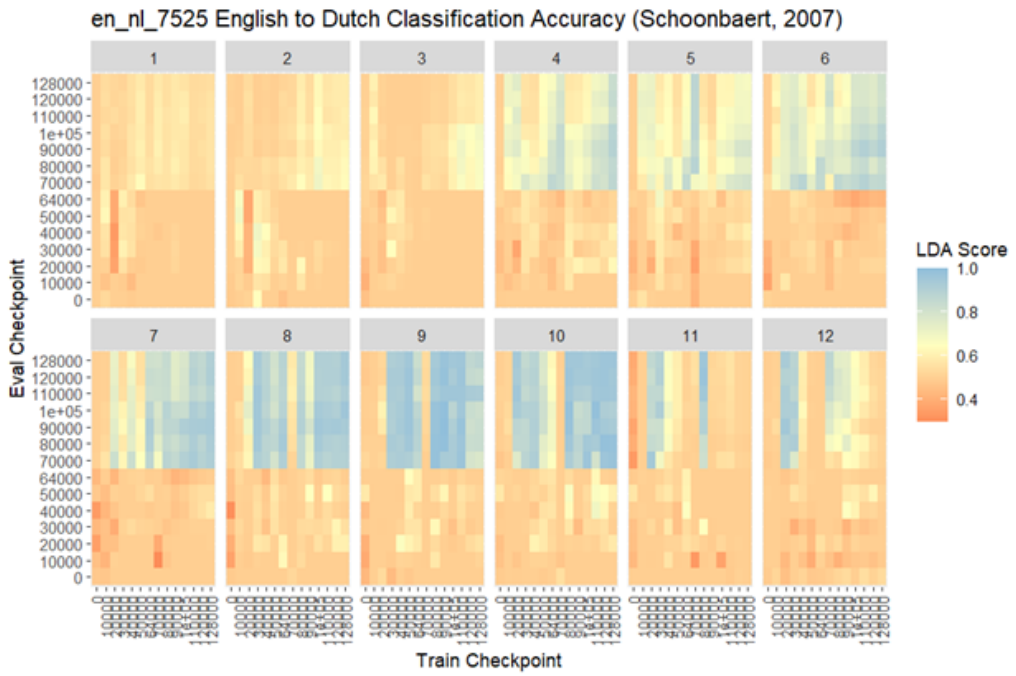

Figure 6.17: Classification accuracy for classifier trained on English activations and evaluated on Dutch activations for Schoonbaert et al. (2007) stimuli for English-Dutch simultaneous model. Each facet represents a layer of the model. Each cell in the figure represents the classification accuracy for a probe trained on a given model checkpoint (x-axis) and evaluated on a given checkpoint (y-axis).

classifying most checkpoints for L2 activations. The representations that the probe identifies in the earlier checkpoints are used to represent L2 through the later stages of training, which suggests that the model uses the same representations from L1 for L2, rather that changing the representation to adapt to L2. This is consistent with the account of transfer learning, where models generalize representations from one context for use in another context. Note, these result are not consistent with zero-shot crosslingual transfer. We only see high accuracy for L2 for the checkpoints after L2 exposure.

For the same model, the LDA scores for the probe trained on L2 activations evaluated on L1 activations show a different pattern (Fig. 6.18. The probes fit on the checkpoints from the first half of training perform at chance for all evaluation checkpoints. This is likely for the same reason as for the L1-L2 results above. The activation patterns for the L2 stimuli do not show any patterns that the probe can use to classify the sentence types before the model has been exposed to L2.

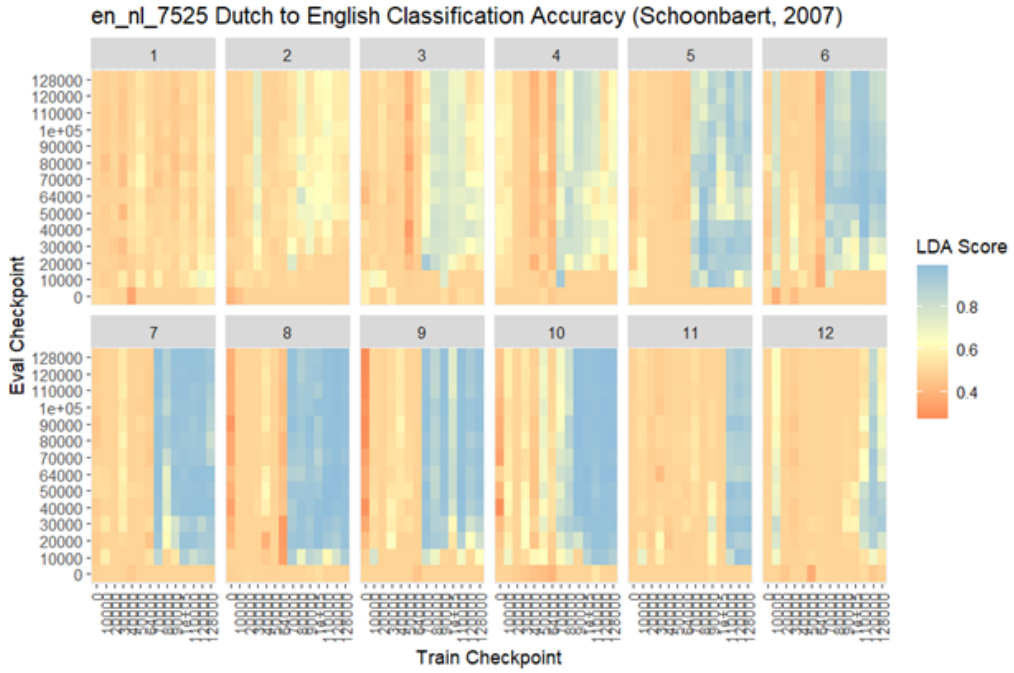

Figure 6.18: Classification accuracy for classifier trained on Dutch activations and evaluated on English activations for Schoonbaert et al. (2007) stimuli for English-Dutch simultaneous model. Each facet represents a layer of the model. Each cell in the figure represents the classification accuracy for a probe trained on a given model checkpoint (x-axis) and evaluated on a given checkpoint (y-axis).

These results also show that the representations identified by the probe are stable over the course of training. The probes trained on on the later checkpoints for

L2 achieve similar accuracy over the evaluation checkpoints, especially for layers 7 to 10. Therefore, the representations for L1 do not change significantly after checkpoint 10000.

For L1-L2 accuracy, the results from models in the sequential bilingual condition show similar results (Fig 6.19). Despite evidence of catastrophic forgetting, there is above-chance classification accuracy from most L1 checkpoints to the L2 checkpoints in the second half. Accuracy overall is lower, especially in the later checkpoints, which is indicated by the color of each cell.

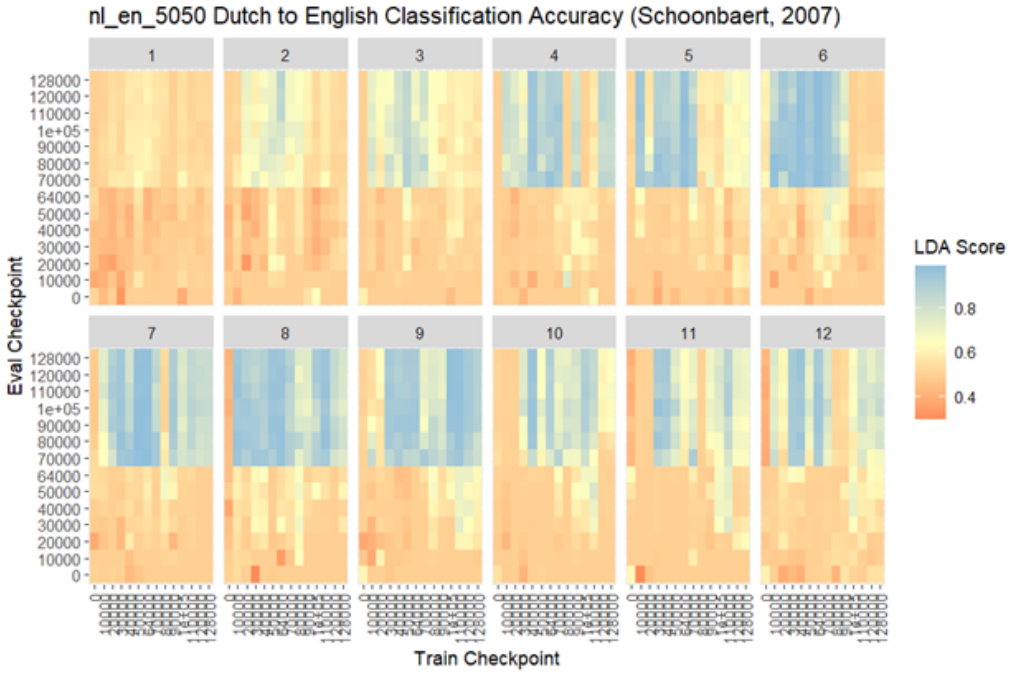

Figure 6.19: Classification accuracy for classifier trained on Dutch activations and evaluated on English activations for Schoonbaert et al. (2007) stimuli for Dutch-English sequential model. Each facet represents a layer of the model. Each cell in the figure represents the classification accuracy for a probe trained on a given model checkpoint (x-axis) and evaluated on a given checkpoint (y-axis).

The L2-L1 plots show a very different pattern in the sequential condition (Fig 6.20). The probe achieves above-chance classification accuracy only for when trained on the later checkpoints on L2 activations and evaluated on the earlier checkpoints on L1 activations. There are similarities between the early L1 representations and the late L2 representations, but not between the representations of the two languages at any other stage.

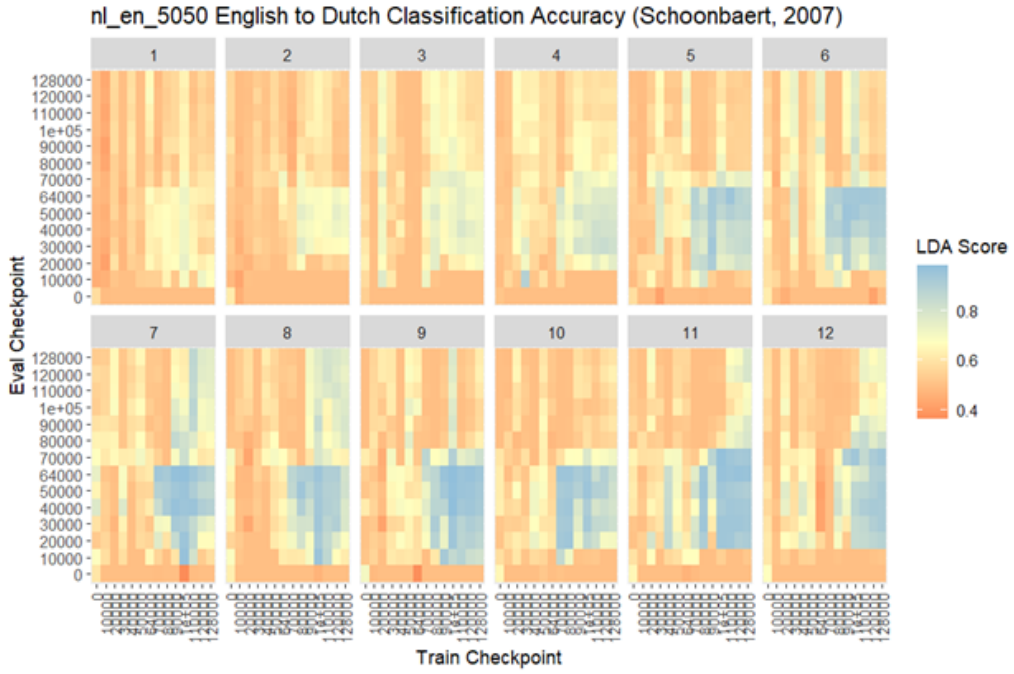

Figure 6.20: Classification accuracy for classifier trained on English activations and evaluated on Dutch activations for Schoonbaert et al. (2007) stimuli for Dutch-English sequential model. Each facet represents a layer of the model. Each cell in the figure represents the classification accuracy for a probe trained on a given model checkpoint (x-axis) and evaluated on a given checkpoint (y-axis).

Together the L1-L2 and L2-L1 results from the sequential bilingual condition show that the catastrophic forgetting involves the transposition L1 and L2 representations, meaning the representations for L1 are used to represent L2 and then representations for L1 drift such that the model no longer is able to represent L1 reliably, but this does not affect the original L1 representations, which are now the L2 representations.

6.6 Causal Analysis

The results from the linear probes are correlational. A causal intervention on the models using the linear probes would demonstrate that the probes have identified the shared abstract grammatical representations we intended to identify. To do this, we used the linear representation to project the activations of each token at each layer. We can shift the activations in the direction of each grammatical alternation along the linear probe. If the probe identified the grammatical representation, the projection should change the relative probability of the grammatical structures.

To do the shifts, we update the activations for each input token according to the mean for one of the grammatical alternations and the linear probe. Given the mean activation μ_{PO} and μ_{DO} for PO and DO sentences respectively, we project the activation to the mean along the LDA axis (v_{LDA}). Specifically, for each input token representation x at layer ℓ, we update[6]:

$$x_{projected,\text{PO}} \leftarrow x + \mu_{\text{PO}} v_{\text{LDA}} \tag{6.3}$$

where:

$$\mu_{\text{PO}} = \langle v_{\text{LDA}}, \mu_{\text{PO}} - x \rangle \tag{6.4}$$

We plot the relative probability of one of the grammatical alternations, PO, in five conditions (Fig. 6.21). The 'no shift' condition is the condition where we do not intervene on the model. In each of the shift conditions, we shift towards either PO or DO. In the conditions where we shifted towards DO, we expect to see

[6] how to indicate einstein summation?

a lower relative probability for PO and in the conditions where we shift towards PO, we expect a higher PO probability. Instead, we see across conditions, there is no significant difference in PO probability. We also test shifts on the probes which were trained using English 'eng') and Dutch 'nld') activations. These results use the stimuli from Schoonbaert et al. (2007) and each of the four Dutch models.

If any of the causal interventions had an effect, the models should assign significantly different probability of PO sentences, relative to the 'no shift' condition; however, there are no significant differences in any conditions.

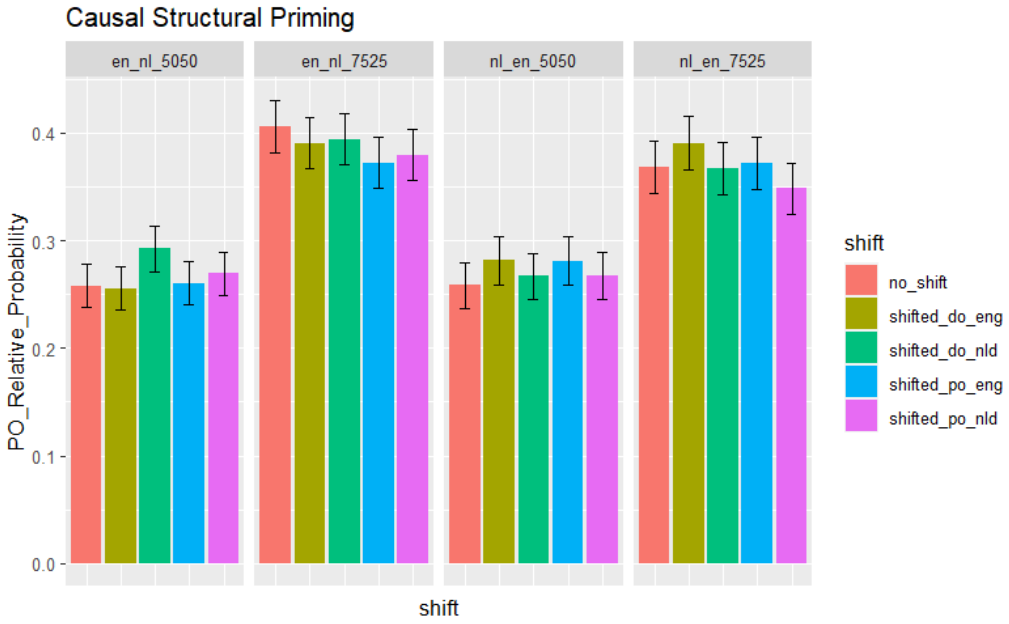

Figure 6.21: Each facet represents the results for each model. From left to right: English-Dutch sequential, English-Dutch simultaneous, Dutch-English sequential, Dutch-English simultaneous. Each bar represents a different shift condition. Stimuli come from Schoonbaert et al. (2007).

We also test the causal intervention with the same method and same set of models, but using the stimuli from Bernolet et al. (2013) (Fig. 6.22). Similarly, we

see no differences between the 'no shift' condition and the conditions where there was a causal intervention.

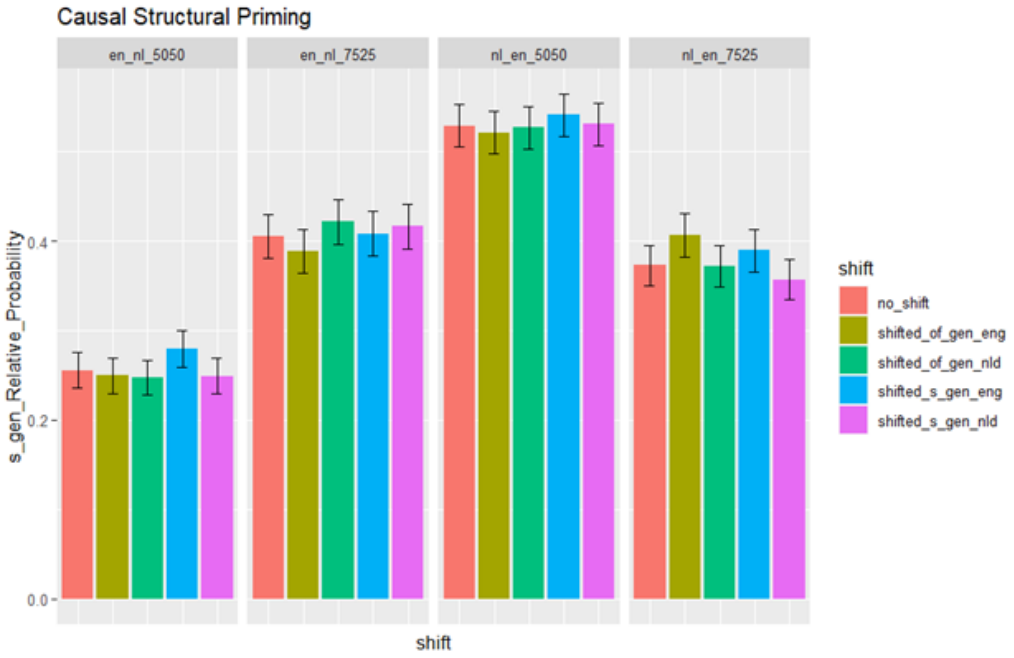

Figure 6.22: Each facet represents the results for each model. From left to right: English-Dutch sequential, English-Dutch simultaneous, Dutch-English sequential, Dutch-English simultaneous. Each bar represents a different shift condition. Stimuli come from Bernolet et al. (2013).

These results are consistent with the linear probe having identified a spurious feature, which is not causally related to the abstract grammatical representations in question.

6.7 What is the linear probe identifying?

The results from the behavioral and probing experiments show that structural priming is driven by shared multilingual abstract representations, which can be identified through a linear probe. However, the causal results show that the linear probe does not have a causal relationship with the abstract grammatical representations.

To determine whether this result is due to the causal intervention or the linear probe itself, we evaluate the selectivity of the linear probe. As the activations are high-dimensional data, it is possible for a linear probe to identify features which may correlate with a surface feature, but not identify the abstract grammatical representation. Under this alternative, we expect to see poor generalization of the probe, which would explain the causal results. If this is the case, we expect to see high classification accuracy, but low selectivity (Hewitt and Liang, 2019). Low selectivity of the linear probe would mean the probes trained on one counterfactual pairs for one grammatical alternation should not be able to reliably classify counterfactual pairs for another alternation.

Figure 6.23 shows LDA classification accuracy across constructions. Each of the plots shows the LDA axis fit to Dutch activations for the DO-PO alternation based on the Schoonbaert et al. (2007) stimuli. Each figure shows the classification accuracy for the English stimuli for the other datasets. In all cases, the two grammatical alternations are linearly separable. We report further results in D.8. There are exceptions for some models and some layers, but these results show that the linear probe lacks selectivity.

Along the LDA axis fit to distinguish DO and PO sentences, of-gen and pas-

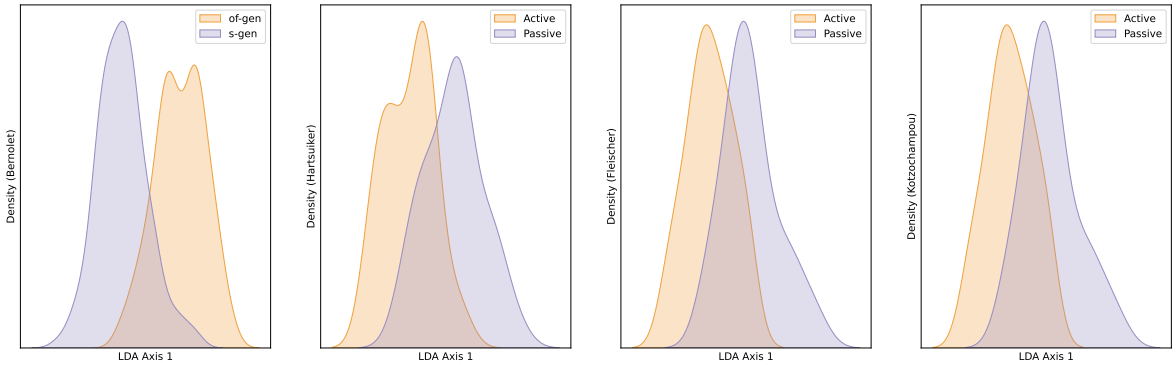

Figure 6.23: English-Dutch simultaneous model trained on Dutch, evaluated on Eng stimuli; Layer 9. From left to right: s-gen/of-gen (Bernolet et al., 2013), active/passive (Hartsuiker et al., 2004), active/passive (Fleischer et al., 2012), active/passive (Kotzochampou and Chondrogianni, 2022)

sive sentences pattern with PO (on the right; e.g. (10)), while s-gen and active sentences pattern with DO (e.g. (11)). There are some similarities between these constructions, which could explain why they pattern together. Understanding the similarities between these constructions could help to understand what representations the probe is identifying.

(10) a. I gave my friend the book (DO)

b. My friend's book (s-gen)

c. I read the book (active)

(11) a. I gave the book to my friend (PO)

b. The book of my friend (of-gen)

c. The book was read by me (passive)

All of the sentences in (10) share argument order. In all three sentences, the agent of the sentence (*I* or *My friend*) precede the theme, *book*. These sentences also all have the animate arguments (*I*, *my friend*) first, followed by the inanimate argument, *book*. All the sentences in (11) generally have reversed argument order and animacy order, relative to (10). The only exception is the DO-PO, as there are three arguments and the order of the subject and agent *I* does not change. All of the sentences in (11) also all contain a preposition (*to*, *of*, *by*).

Based on this, we predict that the sentences in (11) are more likely to prime sentences of different constructions in this same group than their counterparts in (10). We call these groups with-prep and without-prep, respectively. We test for cross-construction structural priming effects. We test both Dutch→English (Fig. 6.24) and English→Dutch (Fig. 6.25) priming. We use the Dutch-English simultaneous and English-Dutch simultaneous models, respectively. We find that in about half of cases, we find cross-construction priming effects. In Dutch→English priming, we see significant priming effects in the expected direction in DO-PO→active-passive and genitive→DO-PO priming. In English→Dutch priming, we see DO-PO→genitive priming. This is indicated by the higher relative probability of the with-prep structure after with with-prep prime (purple) compared to after the without-prep prime (orange).

The causal results suggest that the linear probe learned a spurious representation in the model activations; however, if the probe was identifying a completely meaningless representation, we would not expect to see the probe classify other constructions along linguistically motivated categories. Based on these results, we

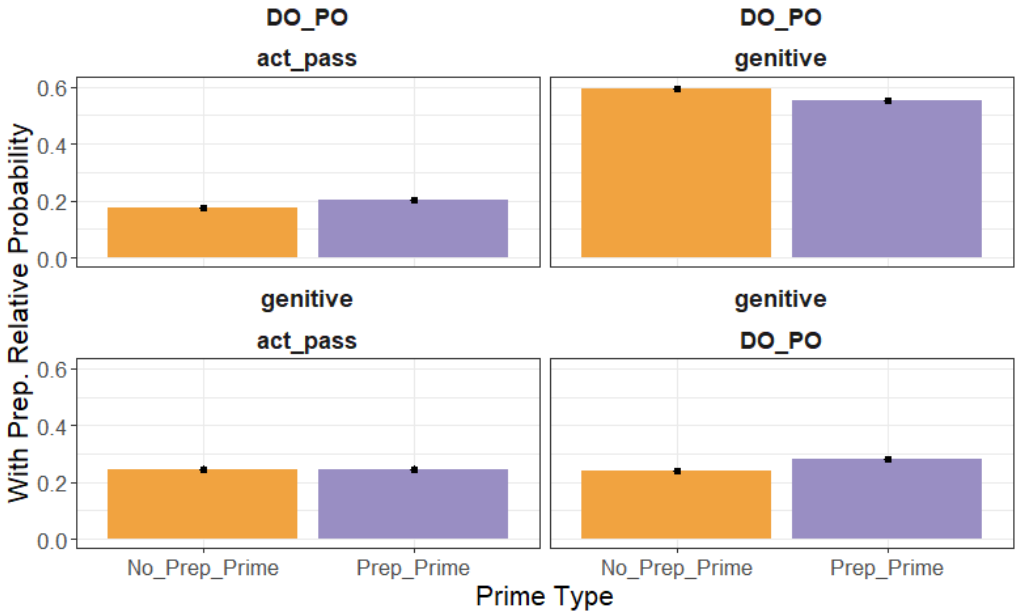

Figure 6.24: Dutch-English simultaneous model, L1-L2

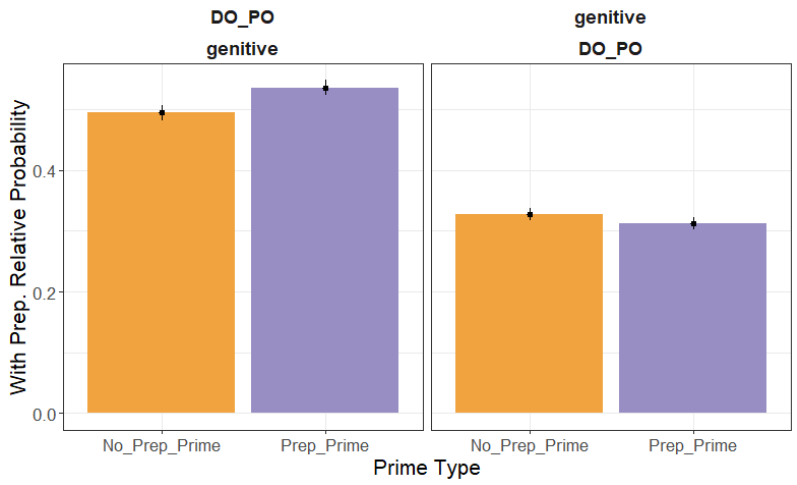

Figure 6.25: English-Dutch simultaneous model, L1-L2

hypothesize that the linear probe is identifying a more general representation that encompasses the three construction pairs.

6.7.1 Alternative Explanations for LDA Results

We discuss three possible types of representations that might be learned by the probe: argument order, presence of a preposition, and sentence length.

Argument order It is unlikely that argument order is driving this effect. In Section 6.3.4, we tested whether word order was driving structural priming effects in Polish. We found that the structural priming effects seemed to be driven by the active-passive distinction, rather than the order of the arguments.

Presence of a preposition To evaluate whether the probes were identifying the presence of a preposition, we created modified stimuli and evaluated the LDA classification. We tested whether adding an additional preposition phrase to both sentences in each pair would affect classification accuracy. We took each of the sentences from Schoonbaert et al. (2007) and added a prepositional phrase modifying the subject of the sentence. In each of the alternations (DO and PO), the added prepositional phrase was in the same place in the sentence. For example, we added the prepositional phrase 'on the boat' to both DO and PO sentences, as shown in (12). We predict that if the linear probe were simply identifying the presence of a preposition, the linear probe should not be able to classify the DO and PO sentences after we added the prepositional phrases.

(12) a. The monk throws the sailor a hat (original DO)

b. The monk **on the boat** throws the the sailor a hat (modified DO)

c. The monk throws a hat to the sailor (original PO)

d. The monk **on the boat** throws a hat to the sailor (modified PO)

Fig. 6.26, we show the LDA accuracy for the original sentences (left) and the sentences with the added prepositional phrases (right). In both cases, when the activations are projected onto the LDA axis, the activations are linearly separable even after adding prepositional phrases. This suggests that the linear probe is not simply identifying the presence of a preposition.

Figure 6.26 shows the classification accuracy for the English-Dutch simultaneous model at layer 7. Additional layer and model results are reported in D.9. Across the models tested and across layers 6 through 10 we find high classification accuracy for both the original and modified stimuli.

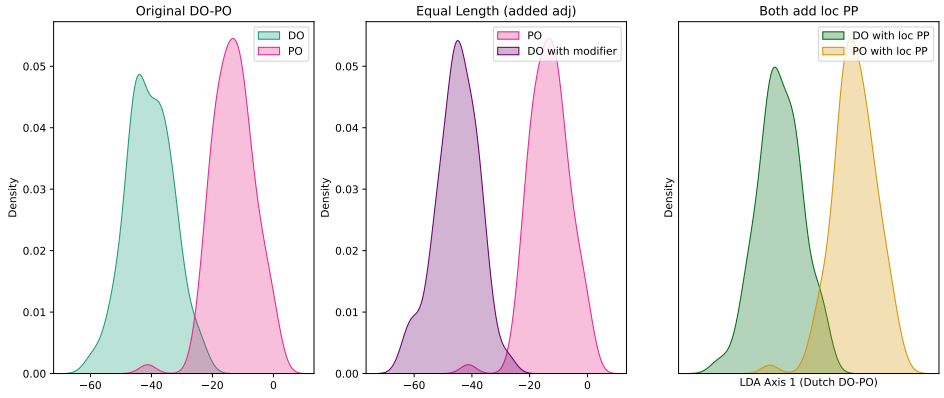

Figure 6.26: Layer 7

Sentence length As the sentences for each alternations are of different lengths – both in terms of orthographic words and tokens – we also test whether the linear probe is classifying the sentences based on the number of tokens. Using the same

logic, we add a modifier to each sentence. Each modifier was added to the direct object, e.g. *hat* in (13). If the sentence length were being used to classify the sentences, we predict that increasing the length of the shorter sentence (in this case DO) should decrease the classification accuracy.

(13) a. The monk throws the sailor a hat (original DO)

b. The monk throws the the sailor a **red** hat (modified DO)

Figure 6.26 shows the classification of the original stimuli (left) compared to the stimuli where all DO sentences have an added modifier (middle). There is no significant change in the classification accuracy of the linear probe. Therefore, it does not seem that the probe is detecting sentence length.

Part II Discussion

In Chapter 5, I provide behavioral evidence for multilingual shared representations through the structural priming experimental paradigm. In Chapter 6, I replicated the structural priming effects from the previous chapter with controlled bilingual language models. Then I describe the training dynamics of structural priming effects, which I showed only emerge after the model has been exposed to both languages. I used a linear probe to classify the activations of the grammatical alternations, and found that the probes are highly accurate at classification; however, I failed to find causal evidence that the representations I identified are linked to structural priming effects. I tested three possible alternative features that the classifier could have been identifying: argument order, presence of a preposition, and sentence length. None of these explains the failure of the causal intervention.

Limitations of Crosslingual Transfer These experiments shed light on both the underlying mechanisms behind and the limitations of crosslingual transfer. There was a marked difference between the robustness and magnitude of structural priming effects for Polish and Greek, relative to Dutch and Spanish. I posit that these are linguistic similarity effects. This not only explains the results from the experiments

in the last two chapters, but is indicative of the general limitations of crosslingual transfer. Even for languages in the same language family (Indo-European), there is still limited ability for models to create successful shared abstract grammatical representations for language pairs such as Greek and English, compared to for a closely related language pair like Dutch and English. Though these experiments are limited only to representations of grammatical structures, and therefore do not bear on shared lexical or conceptual representations, or other kinds of representations such as world knowledge, this suggests the reconsideration of the current practices for leveraging crosslingual transfer.

In Chang et al. (2023a), we show that language relatedness – especially syntactic typological similarity – is predictive of how much benefit there is to adding multilingual data for the improvement performance for a target language, relative to a monolingual setting. Adding data from dissimilar languages always led to less improvement than when added data was from more similar languages. In some settings, dissimilar language data harmed performance, while the similar language data improved or did not change performance, relative to a monolingual baseline.

It is currently common practice to seek to improve performance through adding large quantities of English data, even when the target language is quite different from and unrelated to English. One example of how this approach is applied for a specific language and language model is the representation of Akan in the BLOOM model (Scao et al., 2022). Akan is a language from the Kwa group in the Atlantic-Congo language family. It is the most widely spoken language in Ghana and has at least 9 million native speakers. BLOOM is a language model trained on data from

46 natural languages at several coding languages. There is approximately 0.07MB of Akan data in the training data for BLOOM, which is equivalent to 0.00007% of the training data. Additionally, BLOOM is trained on data from 19 other Niger-Congo languages, making up a total of 0.03378% of the training data (approximately 500MB of data). The vast majority of that data comes from Swahili, which is quite distantly related to Akan within the language family. None of the other Niger-Congo languages represented in BLOOM come from the same language group as Akan, Kwa.

The work in this dissertation, along with previous work, suggests that this may not be an effective method. It is unlikely that a model trained on so little data from related languages could leverage the representations from the majority languages such as English to represent Akan or other Niger-Congo languages. We trained a series of monolingual n-gram language models (Chang et al., 2024) for many low-resource languages like Akan. Using perplexity as an evaluation metric, we found that a small bigram model outperformed BLOOM 7B. This suggests that the level of the Akan performance for BLOOM is worse than that of a bigram model trained on a very small amount of text. Therefore, it seems unlikely that BLOOM effectively leverages crosslingual transfer to boost Akan performance.

Language Asymmetries In human structural priming experiments, it has been shown that structural priming effects are stronger in L1→L2 priming (e.g. Schoonbaert et al., 2007). But the vast majority of these experiments have been conducted with participants for whom the L2 is English. By comparison, very few structural priming experiments have been conducted where English is the L1. Therefore, it is not possible to disentangle whether asymmetries were due to English being the source

versus the target language or due to differences in L1→L2 and L2→L1 priming.

In Chapter 5, we replicated the original L1→L2 human structural priming effects, for which English was the L2, from the original studies. We also tested the reversed language orders, such that English was the prime language. For Schoonbaert et al. (2007) we had human results to compare to, but not for the other language pairs. We also found asymmetrical results, where structural priming effects were stronger when English was the target language. However, there was another confound, which was that all the models were trained on far more English data than data from the other language in the pair. In Chapter 6, I addressed this confound by training controlled models with comparable amounts of language data from each language in a given pair. I tested for structural priming effects with English as the source and target and we also tested for structural priming effects with reversed language orders. The controlled bilingual models still exhibit the asymmetry, where structural priming effects were consistently present for all language pairs when English was the target language, but not always present when English was the prime language. This suggests that there may be an element about structural priming which is stronger in English than in other languages.

This is not easy detect through experimentation with human participants alone, because it is so much easier to find participants for whom English is an L2 than English L1 speakers who speak another language to a very high degree. Therefore, these experiments demonstrate the value of language model experiments in psycholinguistics.

Limitations of Computational Experiments in Psycholinguistics It is not the case that all human experiments can be replace with language model experiments, however. In Chapter 6, I test whether the English asymmetry is due to the higher importance of word order in English, especially compared to languages with case, such as Greek and Polish.

I found that humans and language models demonstrated different effects. Where human structural priming effects seemed sensitive to word order, language model structural priming effects seemed to be driven more by the nature of the grammatical alternation, which was the active-passive alternation in this case. This supports the view that humans and language models use very different mechanisms in language processing. This highlights the limitations of language models as model organisms. While language models may be able simulate some human language behaviors, they may not be well suited to modeling the mechanisms of human language processing. This should constrain, therefore, the types of questions that language models can help address and the types of conclusions that can be drawn from language model experiences. Despite these limitations, the results from language model experiments can be helpful for refining hypotheses, as with the asymmetrical results discussed above.

What Did the Linear Probe Identify? The causal analysis in Chapter 6 did not successfully demonstrate a causal representation between the linear probe and the model outputs. I showed that the linear probe lacked specificity, and was able to classify activation patterns for grammatical alternations other than the ones it was trained to categorize. However, the axis which was tested (DO-PO) classified

the other alternations in a non-random way. The LDA axis classified the sentences consistently in terms of their argument order, although because of the results from the Polish OVS alternation of the passive construction in Section 6.3.4, it seems unlikely that argument order is driving structural priming in language models.

The linear probe also identified that time course of the high classification accuracy was correlated with the model's learning of the L2, as measured by mean surprisal and BLiMP scores. Structural priming effects only emerged as the model learned L2, therefore the shared representations I identified only emerged after sufficient exposure to L2. The linear probe, therefore, had some of the expected traits of the targeted shared representations.

One reason the causal intervention may have failed to affect the model outputs in the expected way is the model activations on which the linear probe was trained. Perhaps the method used to create the sentence representations, SGPT, is not the optimal way of aggregating the contextualized token representations. Perhaps the weighting method obscured information that was needed for the probe to identify the causally linked representations.

It is also possible that fitting the probe to the entire model activations was too coarse of a method. It is possible that fitting a classifier to a more specific part of the activations could lead to a more effective causal intervention. Based on the results, it seems that the probe identified a representation that was too abstract or too general. The probe exhibited a lack of specificity and also did not causally affect the output. In fact, the causal intervention had very little effect on the model outputs. For these reasons, the probe seems too general.

Some work has shown that attention heads track dependencies such as objects of verbs or coreference (Raganato and Tiedemann, 2018; Clark et al., 2019; Mareček and Rosa, 2019; Htut et al., 2019). Perhaps restricting the probe to attention head activations would lead to successful causal intervention.

Chapter 7

Conclusion

In this dissertation, I seek to better understand how language models work differently for different languages and how they work in multilingual settings.

In Part I, I address is to what extent can language model architectures and training recipes be extended from English to new languages. There have been strong claims in the literature that languages cannot be simply applied to any new language (Bender, 2011). Previous work had hypothesized that certain languages, according to certain typological features, would be easier or harder for language models to represent. In Chapter 2, I introduce byte premiums, which indicate differences in dataset size requirements between languages required to convey the same amount of information. In Chapter 4, I replicated the effects and showed that the effects can be accounted for by byte premiums. Difficulties in applying language model architectures to new languages, therefore, may not lie in the architectures themselves but in other parts of the language model ecosystem, such as text encoding. This is

only true, however, after making sure each target language had a custom language-specific tokenizer and each language had a comparable amount of training data. The current language model architectures are very flexible and seem capable of learning every language equally well, but there are other factors that need to be accounted for.

Chapters 3 and 4 provide that morphologically aligned tokenization does not seem necessary for linguistic tasks or explain crosslinguistic differences in language model performance. This challenges a widely held assumption that tokenizers should segment along morpheme boundaries. However, tokenization still remains an understudied aspect of language models, and more work needs to be done to determine what features of tokenizers are most critical for enabling efficient language model performance.

This dissertation also seeks to characterize key aspects of crosslingual transfer, which is the process by which language models generalize across languages. This is a key model feature for improving low-resource language performance, but little is known about how language models learn the shared representations that drive crosslingual transfer. This is the topic of Part II.

In Chapter 5, I use crosslingual structural priming, a paradigm from psycholinguistics, to provide evidence that language models have shared representations for grammatical structures for multiple languages. I argue that these representations are the kind of representations that enable crosslingual transfer, especially for languages that share many syntactic structures. In Chapter 6, I study how language models develop these representations over the course of training. It seems that lan-

guage models can generalize representations to new languages with very little added data, but generalization is most successful when the two languages are similar. This supports other findings, which show that crosslingual transfer is limited to languages that share key features, such as typological similarity.

These studies can help NLP practitioners train multilingual models which more effectively leverage crosslingual transfer to benefit languages for which there is not very much training data available.

Together, the work in this dissertation contributes to ongoing efforts to improve linguistic equity in NLP and work towards more equal language model performance across languages.

Bibliography

Negin Abadani, Jamshid Mozafari, Afsaneh Fatemi, Mohammd Ali Nematbakhsh, and Arefeh Kazemi. Parsquad: machine translated squad dataset for persian question answering. In *2021 7th International Conference on Web Research (ICWR)*, pages 163–168. IEEE, 2021.

Julien Abadji, Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. Ungoliant: An optimized pipeline for the generation of a very large-scale multilingual web corpus. In *Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-9) 2021*, pages 1–9, Mannheim, 2021. Leibniz-Institut für Deutsche Sprache. doi: 10.14618/ids-pub-10468. URL `https://nbn-resolving.org/urn:nbn:de:bsz:mh39-104688`.

Ayimunishagu Abulimiti and Tanja Schultz. Building language models for morphological rich low-resource languages using data from related donor languages: the case of Uyghur. In *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, pages 271–276, Marseille, France, May 2020. European Language Resources association. ISBN 979-10-95546-35-1. URL `https://aclanthology.org/2020.sltu-1.38`.

Farrell Ackerman and Robert Malouf. Morphological organization: The low conditional entropy conjecture. *Language*, 89(3):429–464, 2013.

Ife Adebara, AbdelRahim Elmadany, and Muhammad Abdul-Mageed. Cheetah: Natural language generation for 517 African languages. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12798–12823, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.691. URL `https://aclanthology.org/2024.acl-long.691`.

David Adelani, Hannah Liu, Xiaoyu Shen, Nikita Vassilyev, Jesujoba Alabi, Yanke Mao, Haonan Gao, and En-Shiun Lee. SIB-200: A simple, inclusive, and big evaluation dataset for topic classification in 200+ languages and dialects. In Yvette Graham and Matthew Purver, editors, *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 226–245, St. Julian's, Malta, March 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.eacl-long.14.

Orevaoghene Ahia, Sachin Kumar, Hila Gonen, Jungo Kasai, David Mortensen, Noah Smith, and Yulia Tsvetkov. Do all languages cost the same? tokenization in the era of commercial language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9904–9923, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.614. URL https://aclanthology.org/2023.emnlp-main.614.

Danbi Ahn and Victor S Ferreira. Shared vs separate structural representations: Evidence from cumulative cross-language structural priming. *Quarterly Journal of Experimental Psychology*, 2023. doi: https://doi.org/10.1177/17470218231160942. URL https://journals.sagepub.com/doi/full/10.1177/17470218231160942.

Kabir Ahuja, Sunayana Sitaram, Sandipan Dandapat, and Monojit Choudhury. On the calibration of massively multilingual language models. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4310–4323, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.290. URL https://aclanthology.org/2022.emnlp-main.290.

Rami Al-Rfou, Alex Salcianu, Andy Golding, Anton Bakalov, Chris Alberti, Daniel Andor, David Weiss, Emily Pitler, Greg Coppola, Jason Riesa, Kuzman Ganchev, Michael Ringgaard, Nan Hua, Ryan McDonald, Slav Petrov, Stefan Istrate, and Terry Koo. Gcld3: Cld3 is a neural network model for language identification, 2020. URL https://github.com/google/cld3.

Mehdi Ali, Michael Fromm, Klaudia Thellmann, Richard Rutmann, Max Lübbering, Johannes Leveling, Katrin Klug, Jan Ebert, Niclas Doll, Jasper Buschhoff, Charvi Jain, Alexander Weber, Lena Jurkschat, Hammam Abdelwahab, Chelsea

John, Pedro Ortiz Suarez, Malte Ostendorff, Samuel Weinbach, Rafet Sifa, Stefan Kesselheim, and Nicolas Flores-Herr. Tokenizer choice for LLM training: Negligible or crucial? In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3907–3924, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-naacl.247. URL https://aclanthology.org/2024.findings-naacl.247.

Wazir Ali and Sampo Pyysalo. A survey of large language models for european languages. *arXiv preprint arXiv:2408.15040*, 2024.

Héctor Martínez Alonso and Daniel Zeman. Universal Dependencies for the AnCora treebanks. *Procesamiento del Lenguaje Natural*, 57, 2016. URL https://hal.science/hal-01426751/.

María Angeles Alonso, Angel Fernandez, and Emiliano Díez. Oral frequency norms for 67,979 Spanish words. *Behavior Research Methods*, 43:449–458, 2011. URL https://link.springer.com/article/10.3758/s13428-011-0062-3.

Norah Alzahrani, Hisham Alyahya, Yazeed Alnumay, Sultan AlRashed, Shaykhah Alsubaie, Yousef Almushayqih, Faisal Mirza, Nouf Alotaibi, Nora Al-Twairesh, Areeb Alowisheq, M Saiful Bari, and Haidar Khan. When benchmarks are targets: Revealing the sensitivity of large language model leaderboards. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13787–13805, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.744. URL https://aclanthology.org/2024.acl-long.744.

Aranes, Glyd Jun and Zeman, Dan . Ud cebuano-gja, 2021.

Maria Jesus Aranzabe, Aitziber Atutxa, Kepa Bengoetxea, Koldo Gojenola, and Larraitz Uria. Automatic conversion of the basque dependency treebank to universal dependencies. In *Proceedings of the fourteenth international workshop on treebanks an linguistic theories (TLT14)*, pages 233–241, 2015.

Giorgio Arcara, Carlo Semenza, and Valentina Bambini. Word structure and decomposition effects in reading. *Cognitive Neuropsychology*, 31(1-2):184–218, 2014.

Jordi Armengol-Estapé, Casimiro Pio Carrino, Carlos Rodriguez-Penagos, Ona de Gibert Bonet, Carme Armentano-Oller, Aitor Gonzalez-Agirre, Maite Melero,

and Marta Villegas. Are Multilingual Models the Best Choice for Moderately Under-resourced Languages? A Comprehensive Assessment for Catalan. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4933–4946, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.437. URL https://aclanthology.org/2021.findings-acl.437.

Jordi Armengol-Estapé, Ona de Gibert Bonet, and Maite Melero. On the Multilingual Capabilities of Very Large-Scale English Language Models. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3056–3068, Marseille, France, 2022. European Language Resources Association. URL https://aclanthology.org/2022.lrec-1.327.

Catherine Arnett, Tyler A. Chang, James A. Michaelov, and Benjamin K. Bergen. Crosslingual Structural Priming and the Pre-Training Dynamics of Bilingual Language Models, 2023. URL http://arxiv.org/abs/2310.07929.

Catherine Arnett, Tyler A. Chang, and Benjamin Bergen. A Bit of a Problem: Measurement Disparities in Dataset Sizes across Languages. In Maite Melero, Sakriani Sakti, and Claudia Soria, editors, *Proceedings of the 3rd Annual Meeting of the Special Interest Group on Under-resourced Languages @ LREC-COLING 2024*, pages 1–9, Torino, Italia, May 2024a. ELRA and ICCL. URL https://aclanthology.org/2024.sigul-1.1.

Catherine Arnett, Pamela D Rivière, Tyler Chang, and Sean Trott. Different tokenization schemes lead to comparable performance in Spanish number agreement. In Garrett Nicolai, Eleanor Chodroff, Frederic Mailhot, and Çağrı Çöltekin, editors, *Proceedings of the 21st SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 32–38, Mexico City, Mexico, June 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.sigmorphon-1.4. URL https://aclanthology.org/2024.sigmorphon-1.4.

David Arps, Younes Samih, Laura Kallmeyer, and Hassan Sajjad. Probing for constituency structure in neural language models. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6738–6757, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-emnlp.502. URL https://aclanthology.org/2022.findings-emnlp.502.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, 2020a.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. On the Cross-lingual Transferability of Monolingual Representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online, 2020b. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.421. URL https://aclanthology.org/2020.acl-main.421.

Mohammad Atari, Mona J Xue, Peter S Park, Damián Blasi, and Joseph Henrich. Which humans? 2023. URL https://osf.io/preprints/psyarxiv/5b26t?trk=public_post-text.

R Harald Baayen. Modeling morphological processing. *Morphological aspects of language processing*, 2:257–294, 1995.

Mark Baker. *The polysynthesis parameter*. Oxford Studies in Comparative Syntax. Oxford University Press, 1996.

Lucas Bandarkar, Davis Liang, Benjamin Muller, Mikel Artetxe, Satya Narayan Shukla, Donald Husa, Naman Goyal, Abhinandan Krishnan, Luke Zettlemoyer, and Madian Khabsa. The Belebele benchmark: A parallel reading comprehension dataset in 122 language variants. *arXiv*, 2023. URL https://arxiv.org/abs/2308.16884.

Tamali Banerjee and Pushpak Bhattacharyya. Meaningless yet meaningful: Morphology grounded subword-level NMT. In *Proceedings of the Second Workshop on Subword/Character LEvel Models*, pages 55–60, New Orleans, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-1207. URL https://aclanthology.org/W18-1207.

Douglas M Bates. lme4: Mixed-effects modeling with R, 2010.

Khuyagbaatar Batsuren, Ekaterina Vylomova, Verna Dankers, Tsetsuukhei Delgerbaatar, Omri Uzan, Yuval Pinter, and Gábor Bella. Evaluating subword tokenization: Alien subword composition and oov generalization challenge. *arXiv preprint arXiv:2404.13292*, 2024.

Thomas Bauwens and Pieter Delobelle. BPE-knockout: Pruning pre-existing BPE tokenisers with backwards-compatible morphological semi-supervision. In Kevin

Duh, Helena Gomez, and Steven Bethard, editors, *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5810–5832, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.324. URL https://aclanthology.org/2024.naacl-long.324.

Jatayu Baxi and Brijesh Bhatt. Morpheme boundary detection & grammatical feature prediction for Gujarati : Dataset & model. In Sivaji Bandyopadhyay, Sobha Lalitha Devi, and Pushpak Bhattacharyya, editors, *Proceedings of the 18th International Conference on Natural Language Processing (ICON)*, pages 369–377, National Institute of Technology Silchar, Silchar, India, December 2021. NLP Association of India (NLPAI). URL https://aclanthology.org/2021.icon-main.45.

Emily Bender. The# benderrule: On naming the languages we study and why it matters. *The Gradient*, 14:34, 2019.

Emily M Bender. On achieving and evaluating language-independence in NLP. *Linguistic Issues in Language Technology*, 6, 2011. URL https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=105ed573024e9a31eddc766b6018297ab4383bb9.

Yoav Benjamini and Yosef Hochberg. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):289–300, 1995. ISSN 0035-9246. URL https://www.jstor.org/stable/2346101.

Thomas Berg, Peter Zörnig, and Charlotte Lehr. The effects of type and token frequency on word length: a cross-linguistic study. *Glottotheory*, 13(2): 173–209, 2022. URL https://www.degruyter.com/document/doi/10.1515/glot-2022-2007/html.

Sarah Bernolet, Robert J. Hartsuiker, and Martin J. Pickering. From language-specific to shared syntactic representations: The influence of second language proficiency on syntactic sharing in bilinguals. *Cognition*, 127(3):287–306, 2013. ISSN 0010-0277. doi: 10.1016/j.cognition.2013.02.005. URL https://www.sciencedirect.com/science/article/pii/S0010027713000334.

Mireille Besson, Marta Kutas, and Cyma Van Petten. An Event-Related Potential (ERP) Analysis of Semantic Congruity and Repetition Effects in Sentences. *Jour-*

nal of Cognitive Neuroscience, 4(2):132–149, 1992. ISSN 0898-929X. doi: 10.1162/jocn.1992.4.2.132. URL https://doi.org/10.1162/jocn.1992.4.2.132.

Riyaz Ahmad Bhat, Rajesh Bhatt, Annahita Farudi, Prescott Klassen, Bhuvana Narasimhan, Martha Palmer, Owen Rambow, Dipti Misra Sharma, Ashwini Vaidya, Sri Ramagurumurthy Vishnu, et al. The hindi/urdu treebank project. In *Handbook of Linguistic Annotation*. Springer Press, 2017.

Stella Biderman, Hailey Schoelkopf, Lintang Sutawika, Leo Gao, Jonathan Tow, Baber Abbasi, Alham Fikri Aji, Pawan Sasanka Ammanamanchi, Sidney Black, Jordan Clive, et al. Lessons from the trenches on reproducible evaluation of language models. *arXiv preprint arXiv:2405.14782*, 2024.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439, 2020.

Damian Blasi, Antonios Anastasopoulos, and Graham Neubig. Systematic inequalities in language technology performance across the world's languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5486–5505, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.376. URL https://aclanthology.org/2022.acl-long.376.

James P Blevins. *Word and paradigm morphology*. Oxford University Press, 2016.

Terra Blevins and Luke Zettlemoyer. Language contamination helps explains the cross-lingual capabilities of English pretrained models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3563–3574, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.233. URL https://aclanthology.org/2022.emnlp-main.233.

J. Kathryn Bock. Syntactic persistence in language production. *Cognitive Psychology*, 18(3):355–387, 1986. ISSN 0010-0285. doi: 10.1016/0010-0285(86)90004-6. URL https://www.sciencedirect.com/science/article/pii/0010028586900046.

Rishi Bommasani, Percy Liang, and Tony Lee. Language models are changing ai: The need for holistic evaluation, 2022. URL https://crfm.stanford.edu/2022/11/17/helm.html. Accessed: 2024-12-01.

Kaj Bostrom and Greg Durrett. Byte pair encoding is suboptimal for language model pretraining. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4617–4624, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.414. URL https://aclanthology.org/2020.findings-emnlp.414.

Georgie Botev, Arya D. McCarthy, Winston Wu, and David Yarowsky. Deciphering and characterizing out-of-vocabulary words for morphologically rich languages. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 5309–5326, Gyeongju, Republic of Korea, October 2022. International Committee on Computational Linguistics. URL https://aclanthology.org/2022.coling-1.472.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1075. URL https://aclanthology.org/D15-1075.

Holly P. Branigan and Martin J. Pickering. An experimental approach to linguistic representation. *Behavioral and Brain Sciences*, 40:e282, 2017. ISSN 0140-525X, 1469-1825. doi: 10.1017/S0140525X16002028. URL https://www.cambridge.org/core/journals/behavioral-and-brain-sciences/article/an-experimental-approach-to-linguistic-representation/56398BE6CDD90731063F352A6C65AAB7.

Dunstan Patrick Brown. Morphological typology. In *Handbook of Linguistic Typology*, pages 487–503. Oxford University Press, 2010.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901, 2020. URL https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Brian Butterworth. Lexical representation. *Language production*, 2, 1983.

Joan Bybee. Regular morphology and the lexicon. *Language and cognitive processes*, 10(5):425–455, 1995.

Joan Bybee. *Language, Usage and Cognition*. Cambridge University Press, Cambridge, 2010. ISBN 978-0-521-85140-4. doi: 10.1017/CBO9780511750526. URL https://www.cambridge.org/core/books/language-usage-and-cognition/46BF7213957AF53492A7B03A9BCE9DA0.

Zhenguang G. Cai, Martin J. Pickering, and Holly P. Branigan. Mapping concepts to syntax: Evidence from structural priming in Mandarin Chinese. *Journal of Memory and Language*, 66(4):833–849, 2012. ISSN 0749-596X. doi: 10.1016/j.jml.2012.03.009. URL https://www.sciencedirect.com/science/article/pii/S0749596X12000319.

José Cañete, Gabriel Chaperon, Rodrigo Fuentes, Jou-Hui Ho, Hojin Kang, and Jorge Pérez. Spanish pre-trained BERT model and evaluation data. In *PML4DC at ICLR 2020*, 2020. URL https://arxiv.org/abs/2308.02976.

Yuan Chai, Yaobo Liang, and Nan Duan. Cross-Lingual Ability of Multilingual Masked Language Models: A Study of Language Structure. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4702–4712, Dublin, Ireland, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.322. URL https://aclanthology.org/2022.acl-long.322.

Tyler A Chang and Benjamin K Bergen. Word acquisition in neural language models. *Transactions of the Association for Computational Linguistics*, 10:1–16, 2022.

Tyler A Chang, Zhuowen Tu, and Benjamin K Bergen. The geometry of multilingual language model representations. *arXiv preprint arXiv:2205.10964*, 2022.

Tyler A Chang, Catherine Arnett, Zhuowen Tu, and Benjamin K Bergen. When Is Multilinguality a Curse? Language Modeling for 250 High-and Low-Resource Languages. *arXiv preprint arXiv:2311.09205*, 2023a. URL https://arxiv.org/abs/2311.09205.

Tyler A Chang, Zhuowen Tu, and Benjamin K Bergen. Characterizing learning curves during language model pre-training: Learning, forgetting, and stability. *arXiv preprint arXiv:2308.15419*, 2023b.

Tyler A Chang, Catherine Arnett, Zhuowen Tu, and Benjamin K Bergen. Goldfish: Monolingual Language Models for 350 Languages. *arXiv preprint arXiv:2408.10441*, 2024. URL https://arxiv.org/abs/2408.10441.

Baoguo Chen, Yuefang Jia, Zhu Wang, Susan Dunlap, and Jeong-Ah Shin. Is word-order similarity necessary for cross-linguistic structural priming? *Second Language Research*, 29(4):375–389, 2013. ISSN 0267-6583, 1477-0326. doi: 10.1177/0267658313491962. URL http://journals.sagepub.com/doi/10.1177/0267658313491962.

Edwin Chen. Hellaswag or hellabad? 36% of this popular llm benchmark contains errors, 2024. URL https://www.surgehq.ai/blog/hellaswag-or-hellabad-36-of-this-popular-llm-benchmark-contains-errors. Accessed: 2024-12-01.

Ethan A. Chi, John Hewitt, and Christopher D. Manning. Finding universal grammatical relations in multilingual BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5564–5577, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.493. URL https://aclanthology.org/2020.acl-main.493.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating LLMs by human preference. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=3MW8GKNyzI.

Pavel Chizhov, Catherine Arnett, Elizaveta Korotkova, and Ivan P. Yamshchikov. BPE Gets Picky: Efficient Vocabulary Refinement During Tokenizer Training, 2024. URL https://arxiv.org/abs/2409.04599. Preprint.

Dokook Choe, Rami Al-Rfou, Mandy Guo, Heeyoung Lee, and Noah Constant. Bridging the gap for tokenizer-free language models. *arXiv preprint arXiv:1908.10322*, 2019.

Sunjoo Choi and Myung-Kwan Park. Syntactic priming in the L2 neural language model. *The Journal of Linguistic Science*, 103:81–104, 2022. doi: 10.21296/jls.2022.12.103.81.

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

Noam Chomsky, Ian Roberts, and Jeffrey Watumull. Noam chomsky: The false promise of chatgpt. *The New York Times*, 8, 2023.

Sanghyun Choo and Wonjoon Kim. A study on the evaluation of tokenizer performance in natural language processing. *Applied Artificial Intelligence*, 37(1): 2175112, 2023.

Monojit Choudhury. Generative AI has a language problem. *Nature Human Behaviour*, 7(11):1802–1803, 2023. URL `https://www.nature.com/articles/s41562-023-01716-4`.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

Jayeol Chun, Na-Rae Han, Jena D. Hwang, and Jinho D. Choi. Building Universal Dependency treebanks in Korean. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). URL `https://aclanthology.org/L18-1347`.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4828. URL `https://aclanthology.org/W19-4828`.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. XNLI: Evaluating cross-lingual sentence representations. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1269. URL `https://aclanthology.org/D18-1269`.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online, July 2020a. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.747. URL https://aclanthology.org/2020.acl-main.747.

Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. Emerging cross-lingual structure in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6022–6034, Online, July 2020b. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.536. URL https://aclanthology.org/2020.acl-main.536.

Pablo Contreras Kallens, Ross Deans Kristensen-McLachlan, and Morten H. Christiansen. Large Language Models Demonstrate the Potential of Statistical Learning in Language. *Cognitive Science*, 47(3):e13256, 2023. ISSN 1551-6709. doi: 10.1111/cogs.13256. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/cogs.13256.

Marta R. Costa-jussà, Carlos Escolano, and José A. R. Fonollosa. Byte-based neural machine translation. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, pages 154–158, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4123. URL https://aclanthology.org/W17-4123.

Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. No language left behind: Scaling human-centered machine translation. *arXiv*, 2022. URL https://arxiv.org/abs/2207.04672.

Ryan Cotterell, Sabrina J. Mielke, Jason Eisner, and Brian Roark. Are all languages equally hard to language-model? 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 536–541, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2085. URL https://aclanthology.org/N18-2085.

Lee J Cronbach and Paul E Meehl. Construct validity in psychological tests. *Psychological bulletin*, 52(4):281, 1955.

Maksymilian Dąbkowski and Gašper Beguš. Large language models and (non-) linguistic recursion. *arXiv preprint arXiv:2306.07195*, 2023.

Gautier Dagan, Gabriel Synnaeve, and Baptiste Roziere. Getting the most out of your tokenizer for pre-training and domain adaptation. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=ZFYBnLljtT.

Peter T Daniels. Fundamentals of grammatology. *Journal of the American Oriental Society*, pages 727–731, 1990.

Mark Davis. Unicode over 60 percent of the web, 2012. URL https://googleblog.blogspot.com/2012/02/unicode-over-60-percent-of-web.html. Google Blog.

Andrea Gregor de Varda and Marco Marelli. Data-driven Cross-lingual Syntax: An Agreement Study with Massively Multilingual Models. *Computational Linguistics*, pages 1–39, 01 2023. ISSN 0891-2017. doi: 10.1162/coli_a_00472. URL https://doi.org/10.1162/coli_a_00472.

Björn Deiseroth, Manuel Brack, Patrick Schramowski, Kristian Kersting, and Samuel Weinbach. T-free: Tokenizer-free generative llms via sparse representations for memory-efficient embeddings. *CoRR*, abs/2406.19223, 2024. URL https://doi.org/10.48550/arXiv.2406.19223.

Gregoire Deletang, Anian Ruoss, Paul-Ambroise Duquenne, Elliot Catt, Tim Genewein, Christopher Mattern, Jordi Grau-Moya, Li Kevin Wenliang, Matthew Aitchison, Laurent Orseau, Marcus Hutter, and Joel Veness. Language modeling is compression. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=jznbgiynus.

Gary S. Dell and Victor S. Ferreira. Thirty years of structural priming: An introduction to the special issue. *Journal of Memory and Language*, 91:1–4, 2016. ISSN

0749-596X. doi: 10.1016/j.jml.2016.05.005. URL https://www.sciencedirect.com/science/article/pii/S0749596X16300316.

Chunyuan Deng, Yilun Zhao, Xiangru Tang, Mark Gerstein, and Arman Cohan. Benchmark probing: Investigating data leakage in large language models. In *NeurIPS 2023 Workshop on Backdoors in Deep Learning - The Good, the Bad, and the Ugly*, 2024. URL https://openreview.net/forum?id=a34bgvner1.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Martin d'Hoffschmidt, Wacim Belblidia, Quentin Heinrich, Tom Brendlé, and Maxime Vidal. FQuAD: French question answering dataset. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1193–1208, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.107. URL https://aclanthology.org/2020.findings-emnlp.107.

Guosheng Ding, Danling Peng, and Marcus Taft. The nature of the mental representation of radicals in Chinese: A priming study. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 30(2):530, 2004.

Kaja Dobrovoljc and Nikola Ljubešić. Extending the SSJ Universal Dependencies treebank for Slovenian: Was it worth it? In *Proceedings of the 16th Linguistic Annotation Workshop (LAW-XVI) within LREC2022*, pages 15–22, Marseille, France, June 2022. European Language Resources Association. URL https://aclanthology.org/2022.law-1.3.

Kaja Dobrovoljc, Tomaž Erjavec, and Simon Krek. The Universal Dependencies treebank for Slovenian. In *Proceedings of the 6th Workshop on Balto-Slavic Natural Language Processing*, pages 33–38, Valencia, Spain, April 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-1406. URL https://aclanthology.org/W17-1406.

MM Douglas Bates, Ben Bolker, and Steve Walker. Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, 67(1):1–48, 2015. URL https://www.jstatsoft.org/article/view/v067i01.

Wolfgang U Dressler. A typological approach to first language acquisition. *Language acquisition across linguistic and cognitive systems*, 52:109–124, 2010.

Matthew S. Dryer and Martin Haspelmath. WALS Online (v2020.3), 2013. URL https://wals.info.

Fanny Ducel, Karën Fort, Gaël Lejeune, and Yves Lepage. Do we name the languages we study? the #BenderRule in LREC and ACL articles. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 564–573, Marseille, France, June 2022. European Language Resources Association. URL https://aclanthology.org/2022.lrec-1.60.

eBible. eBible, 2023. URL https://ebible.org/find/.

Daniel Edmiston. A systematic analysis of morphological content in BERT models for multiple languages. *arXiv preprint arXiv:2004.03032*, 2020. URL https://arxiv.org/abs/2004.03032.

Pavel Efimov, Andrey Chertok, Leonid Boytsov, and Pavel Braslavski. Sberquad–russian reading comprehension dataset: Description and analysis. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction: 11th International Conference of the CLEF Association, CLEF 2020, Thessaloniki, Greece, September 22–25, 2020, Proceedings 11*, pages 3–15. Springer, 2020.

Yanai Elazar, Akshita Bhagia, Ian Helgi Magnusson, Abhilasha Ravichander, Dustin Schwenk, Alane Suhr, Evan Pete Walsh, Dirk Groeneveld, Luca Soldaini, Sameer Singh, Hannaneh Hajishirzi, Noah A. Smith, and Jesse Dodge. What's in my big data? In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=RvfPnOkPV4.

Jeffrey L Elman. *Rethinking innateness: A connectionist perspective on development*, volume 10. MIT Press, 1996.

Juuso Eronen, Michal Ptaszynski, and Fumito Masui. Zero-shot cross-lingual transfer language selection using linguistic similarity. *Information Processing & Management*, 60(3):103250, 2023. ISSN 0306-4573. doi: 10.1016/j.ipm.2022.103250. URL https://www.sciencedirect.com/science/article/pii/S030645732200351X.

Martin BH Everaert, Marinus AC Huybregts, Noam Chomsky, Robert C Berwick, and Johan J Bolhuis. Structures, not strings: Linguistics as part of the cognitive sciences. *Trends in cognitive sciences*, 19(12):729–743, 2015.

Gertraud Fenk-Oczlon and August Fenk. Cognition, quantitative linguistics, and systemic typology. *Linguistic Typology*, 3:151–177, 1999.

Victor S. Ferreira and Kathryn Bock. The functions of structural priming. *Language and Cognitive Processes*, 21(7-8):1011–1029, 2006. ISSN 0169-0965. doi: 10.1080/01690960600824609. URL https://doi.org/10.1080/01690960600824609.

JR Firth. A synopsis of linguistic theory 1930-1955. *Studies in Linguistic Analysis, Special Volume/Blackwell*, 1957.

Zuzanna Fleischer, Martin J. Pickering, and Janet F. McLean. Shared information structure: Evidence from cross-linguistic priming. *Bilingualism: Language and Cognition*, 15(3):568–579, 2012. ISSN 1469-1841, 1366-7289. doi: 10.1017/S1366728911000551. URL https://www.cambridge.org/core/journals/bilingualism-language-and-cognition/article/shared-information-structure-evidence-from-crosslinguistic-priming/181577F2E46FB6F02F6850245A610572.

Stefan Frank. Cross-language structural priming in recurrent neural network language models. *Proceedings of the Annual Meeting of the Cognitive Science Society*, 43(43), 2021. URL https://escholarship.org/uc/item/7832j4vp.

Matthias Gallé. Investigating the effectiveness of BPE: The power of shorter sequences. 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 1375–1381, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1141. URL https://aclanthology.org/D19-1141.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The Pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, 07 2024. URL https://zenodo.org/records/12608602.

Jonas Geiping, Alex Stein, Manli Shu, Khalid Saifullah, Yuxin Wen, and Tom Goldstein. Coercing LLMs to do and reveal (almost) anything. In *ICLR*

2024 Workshop on Secure and Trustworthy Large Language Models, 2024. URL https://openreview.net/forum?id=Y5inHAjMu0.

Aryo Pradipta Gema, Joshua Ong Jun Leang, Giwon Hong, Alessio Devoto, Alberto Carlo Maria Mancino, Rohit Saxena, Xuanli He, Yu Zhao, Xiaotang Du, Mohammad Reza Ghasemi Madani, et al. Are we done with mmlu? *arXiv preprint arXiv:2406.04127*, 2024.

Daniela Gerz, Ivan Vulić, Edoardo Ponti, Jason Naradowsky, Roi Reichart, and Anna Korhonen. Language modeling for morphologically rich languages: Character-aware modeling for word-level prediction. *Transactions of the Association for Computational Linguistics*, 6:451–465, 2018a. doi: 10.1162/tacl_a_00032. URL https://aclanthology.org/Q18-1032.

Daniela Gerz, Ivan Vulić, Edoardo Maria Ponti, Roi Reichart, and Anna Korhonen. On the relation between linguistic typology and (limitations of) multilingual language modeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 316–327, Brussels, Belgium, October-November 2018b. Association for Computational Linguistics. doi: 10.18653/v1/D18-1029. URL https://aclanthology.org/D18-1029.

Adele Goldberg. *Constructions at Work: The Nature of Generalization in Language*. Oxford University Press, Oxford, New York, 2006. ISBN 978-0-19-926852-8.

Omer Goldman, Avi Caciularu, Matan Eyal, Kris Cao, Idan Szpektor, and Reut Tsarfaty. Unpacking Tokenization: Evaluating Text Compression and its Correlation with Model Performance. *arXiv preprint arXiv:2403.06265*, 2024. URL https://arxiv.org/pdf/2403.06265.

Charles Goodhart. Problems of monetary management: the uk experience in papers in monetary economics. *Monetary Economics*, 1, 1975.

Adam Goodkind and Klinton Bicknell. Predictive power of word surprisal for reading times is a linear function of language model quality. In *Proceedings of the 8th Workshop on Cognitive Modeling and Computational Linguistics (CMCL 2018)*, pages 10–18, Salt Lake City, Utah, 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-0102. URL https://aclanthology.org/W18-0102.

Edward Gow-Smith, Mark McConville, William Gillies, Jade Scott, and Roibeard Ó Maolalaigh. Use of transformer-based models for word-level transliteration of

the book of the dean of lismore. In *Proceedings of the 4th Celtic Language Technology Workshop within LREC2022*, pages 94–98, Marseille, France, June 2022. European Language Resources Association. URL https://aclanthology.org/2022.cltw-1.13.

Naman Goyal, Jingfei Du, Myle Ott, Giri Anantharaman, and Alexis Conneau. Larger-Scale Transformers for Multilingual Masked Language Modeling. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pages 29–33, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.repl4nlp-1.4. URL https://aclanthology.org/2021.repl4nlp-1.4.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=tEYskw1VY2.

Raffaele Guarasci, Stefano Silvestri, Giuseppe De Pietro, Hamido Fujita, and Massimo Esposito. BERT syntactic transfer: A computational experiment on Italian, French and English languages. *Computer Speech & Language*, 71:101261, 2022. ISSN 0885-2308. doi: 10.1016/j.csl.2021.101261. URL https://www.sciencedirect.com/science/article/pii/S0885230821000681.

Eylon Gueta, Omer Goldman, and Reut Tsarfaty. Explicit Morphological Knowledge Improves Pre-training of Language Models for Hebrew. *arXiv e-prints*, pages arXiv–2311, 2023.

Mandy Guo, Zihang Dai, Denny Vrandečić, and Rami Al-Rfou. Wiki-40B: Multilingual language model dataset. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2440–2452, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://aclanthology.org/2020.lrec-1.297.

Vipul Gupta, David Pantoja, Candace Ross, Adina Williams, and Megan Ung. Changing answer order can decrease mmlu accuracy. *arXiv preprint arXiv:2406.19470*, 2024.

Ximena Gutierrez-Vasques, Christian Bentz, and Tanja Samardžić. Languages through the looking glass of BPE compression. *Computational Linguistics*, 49 (4):943–1001, December 2023. doi: 10.1162/coli_a_00489. URL https://aclanthology.org/2023.cl-4.5.

Dilek Z Hakkani-Tür, Kemal Oflazer, and Gökhan Tür. Statistical morphological disambiguation for agglutinative languages. *Computers and the Humanities*, 36: 381–410, 2002.

Coleman Haley. This is a BERT. Now there are several of them. Can they generalize to novel words? In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 333–341, 2020. URL https://aclanthology.org/2020.blackboxnlp-1.31/.

T Alan Hall. The phonological word: a review. *Studies on the phonological word*, 174(1):22, 1999.

Zellig S. Harris. Distributional Structure. *Word*, 10(2-3):146–162, 1954. doi: 10.1080/00437956.1954.11659520.

Robert J. Hartsuiker, Martin J. Pickering, and Eline Veltkamp. Is Syntax Separate or Shared Between Languages?: Cross-Linguistic Syntactic Priming in Spanish-English Bilinguals. *Psychological Science*, 15(6):409–414, 2004. ISSN 0956-7976. doi: 10.1111/j.0956-7976.2004.00693.x. URL https://doi.org/10.1111/j.0956-7976.2004.00693.x.

Martin Haspelmath. An empirical test of the Agglutination Hypothesis. *Universals of language today*, pages 13–29, 2009. URL https://link.springer.com/chapter/10.1007/978-1-4020-8825-4_2.

Martin Haspelmath. The indeterminacy of word segmentation and the nature of morphology and syntax. *Folia linguistica*, 51(s1000):31–80, 2017.

Martin Haspelmath. Defining the word. *Word*, 69(3):283–297, 2023.

Jonathan Hayase, Alisa Liu, Yejin Choi, Sewoong Oh, and Noah A Smith. Data mixture inference: What do bpe tokenizers reveal about their training data? *arXiv preprint arXiv:2407.16607*, 2024.

Petra Hendriks. *Asymmetries between Language Production and Comprehension*, volume 42 of *Studies in Theoretical Psycholinguistics*. Springer Netherlands, Dordrecht, 2014. ISBN 978-94-007-6900-7 978-94-007-6901-4. doi: 10.1007/978-94-007-6901-4. URL http://link.springer.com/10.1007/978-94-007-6901-4.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021b. URL https://openreview.net/forum?id=7Bywt2mQsCe.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1275. URL https://aclanthology.org/D19-1275.

John Hewitt and Christopher D. Manning. 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, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1419. URL https://aclanthology.org/N19-1419.

Maren Heydel and Wayne S. Murray. Conceptual Effects in Sentence Priming: A Cross-Linguistic Perspective. In Marica De Vincenzi and Vincenzo Lombardo, editors, *Cross-Linguistic Perspectives on Language Processing*, Studies in Theoretical Psycholinguistics, pages 227–254. Springer Netherlands, Dordrecht, 2000. ISBN 978-94-011-3949-6. doi: 10.1007/978-94-011-3949-6_9. URL https://doi.org/10.1007/978-94-011-3949-6_9.

Charles F Hockett. *Refurbishing our foundations*. John Benjamins Publishing Company, 1987.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack W. Rae, and Laurent Sifre. Training compute-optimal large

language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088. URL https://dl.acm.org/doi/abs/10.5555/3600270.3602446.

Valentin Hofmann, Janet Pierrehumbert, and Hinrich Schütze. Superbizarre is not superb: Derivational morphology improves BERT's interpretation of complex words. 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 3594–3608, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.279. URL https://aclanthology.org/2021.acl-long.279.

Valentin Hofmann, Hinrich Schuetze, and Janet Pierrehumbert. An embarrassingly simple method to mitigate undesirable properties of pretrained language model tokenizers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 385–393, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-short.43. URL https://aclanthology.org/2022.acl-short.43.

Nora Hollenstein, Federico Pirovano, Ce Zhang, Lena Jäger, and Lisa Beinborn. Multilingual language models predict human reading behavior. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 106–123, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.10. URL https://aclanthology.org/2021.naacl-main.10.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The Curious Case of Neural Text Degeneration. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rygGQyrFvH.

Yufen Hsieh. Structural priming during sentence comprehension in Chinese–English bilinguals. *Applied Psycholinguistics*, 38(3):657–678, 2017. ISSN 0142-7164, 1469-1817. doi: 10.1017/S0142716416000382.

Phu Mon Htut, Jason Phang, Shikha Bordia, and Samuel R Bowman. Do attention heads in bert track syntactic dependencies? *arXiv preprint arXiv:1911.12246*, 2019.

Jennifer Hu and Michael Frank. Auxiliary task demands mask the capabilities of smaller language models. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=U5BUzSn4tD.

Jennifer Hu and Roger Levy. Prompt-based methods may underestimate large language models' linguistic generalizations. *arXiv preprint arXiv:2305.13264*, 2023.

Ayyoob Imani, Peiqin Lin, Amir Hossein Kargaran, Silvia Severini, Masoud Jalili Sabet, Nora Kassner, Chunlan Ma, Helmut Schmid, André Martins, François Yvon, and Hinrich Schütze. Glot500: Scaling multilingual corpora and language models to 500 languages. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1082–1117, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.61. URL https://aclanthology.org/2023.acl-long.61.

Haris Jabbar. MorphPiece: A linguistic tokenizer for large language models. *arXiv*, 2024. URL https://arxiv.org/pdf/2307.07262.pdf.

Abigail Z Jacobs and Hanna Wallach. Measurement and fairness. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 375–385, 2021.

Shailee Jain, Vy A Vo, Leila Wehbe, and Alexander G Huth. Computational language modeling and the promise of in silico experimentation. *Neurobiology of Language*, 5(1):80–106, 2024.

Matias Jentoft and David Samuel. NoCoLA: The Norwegian corpus of linguistic acceptability. In Tanel Alumäe and Mark Fishel, editors, *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 610–617, Tórshavn, Faroe Islands, May 2023. University of Tartu Library. URL https://aclanthology.org/2023.nodalida-1.60.

Taehee Jeon, Bongseok Yang, Changhwan Kim, and Yoonseob Lim. Improving korean nlp tasks with linguistically informed subword tokenization and sub-character decomposition. *arXiv preprint arXiv:2311.03928*, 2023.

Marc F Joanisse and James L McClelland. Connectionist perspectives on language learning, representation and processing. *Wiley Interdisciplinary Reviews: Cognitive Science*, 6(3):235–247, 2015.

Alexander Jones, William Yang Wang, and Kyle Mahowald. A Massively Multilingual Analysis of Cross-linguality in Shared Embedding Space. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5833–5847, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.471. URL https://aclanthology.org/2021.emnlp-main.471.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.560. URL https://aclanthology.org/2020.acl-main.560.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain, April 2017. Association for Computational Linguistics. URL https://aclanthology.org/E17-2068.

Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models*. Online manuscript, 3rd edition, 2024. URL https://web.stanford.edu/~jurafsky/slp3/3.pdf. Online manuscript released August 20, 2024.

Karthikeyan K, Zihan Wang, Stephen Mayhew, and Dan Roth. Cross-Lingual Ability of Multilingual BERT: An Empirical Study. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HJeT3yrtDr&utm_campaign=NLP%20News&utm_medium=email&utm_source=Revue%20newsletter.

Julie Kallini, Isabel Papadimitriou, Richard Futrell, Kyle Mahowald, and Christopher Potts. Mission: Impossible language models. *arXiv preprint arXiv:2401.06416*, 2024.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling Laws for Neural Language Models, 2020. URL http://arxiv.org/abs/2001.08361.

Yiğit Bekir Kaya and A Cüneyd Tantuğ. Effect of tokenization granularity for Turkish large language models. *Intelligent Systems with Applications*, 21: 200335, 2024. URL https://www.sciencedirect.com/science/article/pii/S2667305324000115.

Simran Khanuja, Sebastian Ruder, and Partha Talukdar. Evaluating the diversity, equity, and inclusion of NLP technology: A case study for Indian languages. In Andreas Vlachos and Isabelle Augenstein, editors, *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1763–1777, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-eacl.131. URL https://aclanthology.org/2023.findings-eacl.131.

Christo Kirov, Ryan Cotterell, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sabrina J. Mielke, Arya McCarthy, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. UniMorph 2.0: Universal Morphology. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). URL https://aclanthology.org/L18-1293.

Stav Klein and Reut Tsarfaty. Getting the ##life out of living: How Adequate Are Word-Pieces for Modelling Complex Morphology? In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 204–209, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.sigmorphon-1.24. URL https://www.aclweb.org/anthology/2020.sigmorphon-1.24.

Sotiria Kotzochampou and Vasiliki Chondrogianni. How similar are shared syntactic representations? Evidence from priming of passives in Greek–English bilinguals. *Bilingualism: Language and Cognition*, 25(5):726–738, 2022. ISSN 1366-7289, 1469-1841. doi: 10.1017/S136672892200027X.

Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, et al. Quality at a glance: An audit of web-crawled multilingual datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72, 2022.

Daria Kryvosheieva and Roger Levy. Controlled evaluation of syntactic knowledge in multilingual language models. *arXiv preprint arXiv:2411.07474*, 2024.

Taku Kudo and John Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium, 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-2012. URL https://aclanthology.org/D18-2012.

Sneha Kudugunta, Isaac Caswell, Biao Zhang, Xavier Garcia, Derrick Xin, Aditya Kusupati, Romi Stella, Ankur Bapna, and Orhan Firat. Madlad-400: A multilingual and document-level large audited dataset. *Advances in Neural Information Processing Systems*, 36, 2024.

Tatsuki Kuribayashi, Yohei Oseki, Takumi Ito, Ryo Yoshida, Masayuki Asahara, and Kentaro Inui. Lower Perplexity is Not Always Human-Like. 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 5203–5217, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.405. URL https://aclanthology.org/2021.acl-long.405.

Marta Kutas and Steven A Hillyard. Brain potentials during reading reflect word expectancy and semantic association. *Nature*, 307(5947):161–163, 1984.

Sander Land and Max Bartolo. Fishing for magikarp: Automatically detecting undertrained tokens in large language models. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11631–11646, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.649. URL https://aclanthology.org/2024.emnlp-main.649.

Septina Dian Larasati, Vladislav Kuboň, and Daniel Zeman. Indonesian morphology tool (morphind): Towards an indonesian corpus. In *Systems and Frameworks for Computational Morphology: Second International Workshop, SFCM 2011, Zurich, Switzerland, August 26, 2011. Proceedings 2*, pages 119–129. Springer, 2011.

Jungseob Lee, Hyeonseok Moon, Seungjun Lee, Chanjun Park, Sugyeong Eo, Hyunwoong Ko, Jaehyung Seo, Seungyoon Lee, and Heuiseok Lim. Length-aware byte pair encoding for mitigating over-segmentation in Korean machine translation. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics ACL 2024*, pages 2287–2303, Bangkok,

Thailand and virtual meeting, August 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.findings-acl.135.

Bai Li, Zining Zhu, Guillaume Thomas, Frank Rudzicz, and Yang Xu. Neural reality of argument structure constructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7410–7423, Dublin, Ireland, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.512. URL https://aclanthology.org/2022.acl-long.512.

Davis Liang, Hila Gonen, Yuning Mao, Rui Hou, Naman Goyal, Marjan Ghazvininejad, Luke Zettlemoyer, and Madian Khabsa. XLM-V: Overcoming the Vocabulary Bottleneck in Multilingual Masked Language Models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13142–13152, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.813. URL https://aclanthology.org/2023.emnlp-main.813.

Jindřich Libovickỳ and Jindřich Helcl. Lexically grounded subword segmentation. *arXiv preprint arXiv:2406.13560*, 2024.

Noémi Ligeti-Nagy, Gergő Ferenczi, Enikő Héja, László János Laki, Noémi Vadász, Zijian Győző Yang, and Tamás Váradi. HuLU: Hungarian language understanding benchmark kit. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 8360–8371, Torino, Italia, May 2024. ELRA and ICCL. URL https://aclanthology.org/2024.lrec-main.733.

Tomasz Limisiewicz, Terra Blevins, Hila Gonen, Orevaoghene Ahia, and Luke Zettlemoyer. MYTE: Morphology-driven byte encoding for better and fairer multilingual language modeling. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15059–15076, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.804. URL https://aclanthology.org/2024.acl-long.804.

Peiqin Lin, Shaoxiong Ji, Jörg Tiedemann, André FT Martins, and Hinrich Schütze. MaLA-500: Massive Language Adaptation of Large Language Models. *arXiv preprint arXiv:2401.13303*, 2024. URL https://arxiv.org/abs/2401.13303.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona Diab, Veselin Stoyanov, and Xian Li. Few-shot learning with multilingual generative language models. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9019–9052, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.616. URL https://aclanthology.org/2022.emnlp-main.616.

Yongjie Lin, Yi Chern Tan, and Robert Frank. Open sesame: Getting inside BERT's linguistic knowledge. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 241–253, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4825. URL https://aclanthology.org/W19-4825.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. Assessing the ability of lstms to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535, 2016.

Yikang Liu, Yeting Shen, Hongao Zhu, Lilong Xu, Zhiheng Qian, Siyuan Song, Kejia Zhang, Jialong Tang, Pei Zhang, Baosong Yang, et al. Zhoblimp: a systematic assessment of language models with linguistic minimal pairs in chinese. *arXiv preprint arXiv:2411.06096*, 2024.

Irina Lobzhanidze. *Finite-State Computational Morphology.*

Helga Loebell and Kathryn Bock. Structural priming across languages. *Linguistics*, 41(5):791–824, 2003. ISSN 1613-396X. doi: 10.1515/ling.2003.026. URL https://www.degruyter.com/document/doi/10.1515/ling.2003.026/html.

Shayne Longpre, Robert Mahari, Ariel N. Lee, Campbell S. Lund, Hamidah Oderinwale, William Brannon, Nayan Saxena, Naana Obeng-Marnu, Tobin South, Cole J Hunter, Kevin Klyman, Christopher Klamm, Hailey Schoelkopf, Nikhil Singh, Manuel Cherep, Ahmad Mustafa Anis, An Dinh, Caroline Shamiso Chitongo, Da Yin, Damien Sileo, Deividas Mataciunas, Diganta Misra, Emad A. Alghamdi, Enrico Shippole, Jianguo Zhang, Joanna Materzynska, Kun Qian, Kushagra Tiwary, Lester James Validad Miranda, Manan Dey, Minnie Liang, Mohammed Hamdy, Niklas Muennighoff, Seonghyeon Ye, Seungone Kim, Shrestha

Mohanty, Vipul Gupta, Vivek Sharma, Vu Minh Chien, Xuhui Zhou, Yizhi LI, Caiming Xiong, Luis Villa, Stella Biderman, Hanlin Li, Daphne Ippolito, Sara Hooker, Jad Kabbara, and Alex Pentland. Consent in crisis: The rapid decline of the AI data commons. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=66PcEzkf95.

F. G. Lounsbury. *Oneida Verb Morphology*. Yale University Press, Dept. of Anthropology, Yale University, 1953.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.556. URL https://aclanthology.org/2022.acl-long.556.

Teresa Lynn and Jennifer Foster. Universal Dependencies for Irish. In *Proceedings of the Second Celtic Language Technology Workshop*, Paris, France, 2016.

Bill MacCartney and Christopher D. Manning. Modeling semantic containment and exclusion in natural language inference. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 521–528, Manchester, UK, August 2008. Coling 2008 Organizing Committee. URL https://aclanthology.org/C08-1066.

Dominik Macháček, Jonáš Vidra, and Ondřej Bojar. Morphological and language-agnostic word segmentation for NMT. In *International Conference on Text, Speech, and Dialogue*, pages 277–284. Springer, 2018.

Kyle Mahowald, Ariel James, Richard Futrell, and Edward Gibson. A meta-analysis of syntactic priming in language production. *Journal of Memory and Language*, 91:5–27, 2016. ISSN 0749-596X. doi: 10.1016/j.jml.2016.03.009. URL https://www.sciencedirect.com/science/article/pii/S0749596X16300043.

Kyle Mahowald, Anna A Ivanova, Idan A Blank, Nancy Kanwisher, Joshua B Tenenbaum, and Evelina Fedorenko. Dissociating language and thought in large language models. *Trends in Cognitive Sciences*, 2024.

David Mareček and Rudolf Rosa. From balustrades to pierre vinken: Looking for syntax in transformer self-attentions. In *Proceedings of the 2019 ACL Workshop*

BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, pages 263–275, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4827. URL https://aclanthology.org/W19-4827.

Clara D. Martin, Francesca M. Branzi, and Moshe Bar. Prediction is Production: The missing link between language production and comprehension. *Scientific Reports*, 8(1):1079, 2018. ISSN 2045-2322. doi: 10.1038/s41598-018-19499-4. URL https://www.nature.com/articles/s41598-018-19499-4.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. CamemBERT: a tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.645. URL https://aclanthology.org/2020.acl-main.645.

Kosuke Matsuzaki, Masaya Taniguchi, Kentaro Inui, and Keisuke Sakaguchi. J-UniMorph: Japanese morphological annotation through the universal feature schema. In Garrett Nicolai, Eleanor Chodroff, Frederic Mailhot, and Çağrı Çöltekin, editors, *Proceedings of the 21st SIGMORPHON workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 7–19, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.sigmorphon-1.2. URL https://aclanthology.org/2024.sigmorphon-1.2.

Rowan Hall Maudslay and Ryan Cotterell. Do syntactic probes probe syntax? experiments with jabberwocky probing. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 124–131, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.11. URL https://aclanthology.org/2021.naacl-main.11.

Stephen Mayhew, Terra Blevins, Shuheng Liu, Marek Suppa, Hila Gonen, Joseph Marvin Imperial, Börje Karlsson, Peiqin Lin, Nikola Ljubešić, Lester James Miranda, Barbara Plank, Arij Riabi, and Yuval Pinter. Universal NER: A gold-standard multilingual named entity recognition benchmark. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4322–4337, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.243. URL https://aclanthology.org/2024.naacl-long.243.

Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989.

Meta AI. Llama 3.1 8b model card, 2024. URL `https://huggingface.co/meta-llama/Llama-3.1-8B`. Accessed: 2024-12-02.

Antje S. Meyer, Falk Huettig, and Willem J.M. Levelt. Same, different, or closely related: What is the relationship between language production and comprehension? *Journal of Memory and Language*, 89:1–7, 2016. ISSN 0749596X. doi: 10.1016/j.jml.2016.03.002. URL `https://linkinghub.elsevier.com/retrieve/pii/S0749596X16000243`.

James Michaelov and Ben Bergen. The more human-like the language model, the more surprisal is the best predictor of n400 amplitude. In *NeurIPS 2022 Workshop on Information-Theoretic Principles in Cognitive Systems*, 2022a.

James Michaelov, Catherine Arnett, and Ben Bergen. Revenge of the fallen? recurrent models match transformers at predicting human language comprehension metrics. In *First Conference on Language Modeling*, 2024. URL `https://openreview.net/forum?id=amhPBLFYWv`.

James A. Michaelov and Benjamin K. Bergen. Do language models make human-like predictions about the coreferents of Italian anaphoric zero pronouns? In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1–14, Gyeongju, Republic of Korea, October 2022b. International Committee on Computational Linguistics. URL `https://aclanthology.org/2022.coling-1.1`.

James A. Michaelov and Benjamin K. Bergen. Do language models make human-like predictions about the coreferents of italian anaphoric zero pronouns? *arXiv preprint arXiv:2208.14554*, 2022c.

James A Michaelov and Benjamin K Bergen. Ignoring the alternatives: The n400 is sensitive to stimulus preactivation alone. *cortex*, 168:82–101, 2023.

James A. Michaelov, Seana Coulson, and Benjamin K. Bergen. So Cloze yet so Far: N400 Amplitude is Better Predicted by Distributional Information than Human Predictability Judgements. *IEEE Transactions on Cognitive and Developmental Systems*, 2022. ISSN 2379-8939. doi: 10.1109/TCDS.2022.3176783.

Vladislav Mikhailov, Tatiana Shamardina, Max Ryabinin, Alena Pestova, Ivan Smurov, and Ekaterina Artemova. RuCoLA: Russian corpus of linguistic acceptability. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5207–5227, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.348. URL https://aclanthology.org/2022.emnlp-main.348.

Tomas Mikolov. Distributed representations of words and phrases and their compositionality. *arXiv preprint arXiv:1310.4546*, 2013a.

Tomas Mikolov. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 3781, 2013b.

Evan Miller. Adding error bars to evals: A statistical approach to language model evaluations. *arXiv preprint arXiv:2411.00640*, 2024.

Raphaël Millière. Language models as models of language. In R. Nefdt, G. Dupre, and K. Stanton, editors, *The Oxford Handbook of the Philosophy of Linguistics*. Oxford University Press, forthcoming. Forthcoming.

Benjamin Minixhofer, Jonas Pfeiffer, and Ivan Vulić. CompoundPiece: Evaluating and improving decompounding performance of language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 343–359, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.24. URL https://aclanthology.org/2023.emnlp-main.24.

Kanishka Misra and Najoung Kim. Generating novel experimental hypotheses from language models: A case study on cross-dative generalization. *arXiv preprint arXiv:2408.05086*, 2024.

Kanishka Misra and Kyle Mahowald. Language models learn rare phenomena from less rare phenomena: The case of the missing AANNs. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 913–929, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.53. URL https://aclanthology.org/2024.emnlp-main.53.

Penny F. Mitchell, Sally Andrews, and Philip B. Ward. An event-related potential study of semantic congruity and repetition in a sentence-reading task: Effects of context change. *Psychophysiology*, 30(5):496–509, 1993. ISSN 1469-8986. doi: 10.1111/j.1469-8986.1993.tb02073.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-8986.1993.tb02073.x.

Timo Möller, Julian Risch, and Malte Pietsch. GermanQuAD and GermanDPR: Improving non-English question answering and passage retrieval. In *Proceedings of the 3rd Workshop on Machine Reading for Question Answering*, pages 42–50, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.mrqa-1.4. URL https://aclanthology.org/2021.mrqa-1.4.

Steven Moran, Daniel McCloy, and Richard Wright. PHOIBLE online, 2014. URL https://phoible.org/.

Edith A Moravcsik. *Introducing language typology*. Cambridge University Press, 2012.

Aaron Mueller and Tal Linzen. How to plant trees in language models: Data and architectural effects on the emergence of syntactic inductive biases. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11237–11252, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.629. URL https://aclanthology.org/2023.acl-long.629.

Aaron Mueller, Garrett Nicolai, Panayiota Petrou-Zeniou, Natalia Talmina, and Tal Linzen. Cross-linguistic syntactic evaluation of word prediction models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5523–5539, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.490. URL https://aclanthology.org/2020.acl-main.490.

Aaron Mueller, Robert Frank, Tal Linzen, Luheng Wang, and Sebastian Schuster. Coloring the blank slate: Pre-training imparts a hierarchical inductive bias to sequence-to-sequence models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1352–1368, Dublin, Ireland, May 2022a. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.106. URL https://aclanthology.org/2022.findings-acl.106.

Aaron Mueller, Yu Xia, and Tal Linzen. Causal analysis of syntactic agreement neurons in multilingual language models. In Antske Fokkens and Vivek Srikumar, editors, *Proceedings of the 26th Conference on Computational Natural Language Learning (CoNLL)*, pages 95–109, Abu Dhabi, United Arab Emirates (Hybrid), December 2022b. Association for Computational Linguistics. doi: 10.18653/v1/2022.conll-1.8. URL https://aclanthology.org/2022.conll-1.8.

Niklas Muennighoff. SGPT: GPT Sentence Embeddings for Semantic Search. *arXiv preprint arXiv:2202.08904*, 2022.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. Crosslingual Generalization through Multitask Finetuning. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.891. URL https://aclanthology.org/2023.acl-long.891.

Stefan Müller. Large language models: The best linguistic theory, a wrong linguistic theory, or no linguistic theory at all? *Lingbuzz Preprint*, 2024.

Merel Muylle, Sarah Bernolet, and Robert J. Hartsuiker. The Role of Case Marking and Word Order in Cross-Linguistic Structural Priming in Late L2 Acquisition. *Language Learning*, 70(S2):194–220, 2020. ISSN 1467-9922. doi: 10.1111/lang.12400. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/lang.12400.

Daniel Nettle. Segmental inventory size, word length, and communicative efficiency. *Linguistics*, 33(2):359–367, 1995.

Hellina Hailu Nigatu, John Canny, and Sarah Chasins. A need finding study with low-resource language content creators. In *Proceedings of the 4th African Human Computer Interaction Conference*, pages 1–4, 2023.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi

Akula, Loic Barrault, Gabriel Mejia-Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. No language left behind: Scaling human-centered machine translation. 2022.

Javad Nouri and Roman Yangarber. A novel evaluation method for morphological segmentation. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3102–3109, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA). URL `https://aclanthology.org/L16-1495`.

Antoine Nzeyimana and Andre Niyongabo Rubungo. KinyaBERT: A Morphology-aware Kinyarwanda Language Model. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5347–5363, Dublin, Ireland, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.367. URL `https://aclanthology.org/2022.acl-long.367`.

Occiglot. Eu tokenizer performance. `https://occiglot.eu/posts/eu_tokenizer_perfomance/`, 2024. Accessed: Nov. 17, 2024.

Kelechi Ogueji, Yuxin Zhu, and Jimmy Lin. Small Data? No Problem! Exploring the Viability of Pretrained Multilingual Language Models for Low-resourced Languages. In *Proceedings of the 1st Workshop on Multilingual Representation Learning*, pages 116–126, Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.mrl-1.11. URL `https://aclanthology.org/2021.mrl-1.11`.

Byung-Doh Oh and William Schuler. Why does surprisal from larger transformer-based language models provide a poorer fit to human reading times? *Transactions of the Association for Computational Linguistics*, 11:336–350, 2023.

Akintunde Oladipo, Odunayo Ogundepo, Kelechi Ogueji, and Jimmy Lin. An exploration of vocabulary size and transfer effects in multilingual language models for african languages. In *3rd Workshop on African Natural Language Processing*, 2022.

OpenAI. Learning to reason with large language models, 2024. URL https://web.archive.org/web/20241118063143/https://openai.com/index/learning-to-reason-with-llms/. Accessed: 2024-11-18.

Artidoro Pagnoni, Ram Pasunuru, Pedro Rodriguez, John Nguyen, Benjamin Muller, Margaret Li, Chunting Zhou, Lili Yu, Jason Weston, Luke Zettlemoyer, Gargi Ghosh, Mike Lewis, Ari Holtzman, and Srinivasan Iyer. Byte latent transformer: Patches scale better than tokens, 2024. URL https://scontent-bos5-1.xx.fbcdn.net/v/t39.2365-6/470135129_1314438233309836_4712217603129928862_n.pdf?_nc_cat=111&ccb=1-7&_nc_sid=3c67a6&_nc_ohc=_rrHgJZzzMcQ7kNvgEdSo4C&_nc_zt=14&_nc_ht=scontent-bos5-1.xx&_nc_gid=A6O8eGZbLdGznY4r29rgHlM&oh=00_AYD0uSPzWpWXwgYt23nbS5I8OWQfK3EFS85BwLHQyfBfVw&oe=67623EC8. Preprint.

Martha Palmer, Rajesh Bhatt, Bhuvana Narasimhan, Owen Rambow, Dipti Misra Sharma, and Fei Xia. Hindi syntax: Annotating dependency, lexical predicate-argument structure, and phrase structure. In *The 7th International Conference on Natural Language Processing*, pages 14–17, 2009.

Chanjun Park, Sugyeong Eo, Hyeonseok Moon, and Heuiseok Lim. Should we find another model?: Improving Neural Machine Translation Performance with ONE-Piece Tokenization Method without Model Modification. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Papers*, pages 97–104, Online, 2021a. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-industry.13. URL https://aclanthology.org/2021.naacl-industry.13.

Hyunji Hayley Park, Katherine J Zhang, Coleman Haley, Kenneth Steimel, Han Liu, and Lane Schwartz. Morphology matters: A multilingual language modeling analysis. *Transactions of the Association for Computational Linguistics*, 9:261–276, 2021b.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=UGpGkLzwpP.

Kyubyong Park, Joohong Lee, Seongbo Jang, and Dawoon Jung. An empirical study of tokenization strategies for various Korean NLP tasks. In *Proceedings of the 1st*

Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing, pages 133–142, Suzhou, China, December 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.aacl-main.17.

Abhinav Patil, Jaap Jumelet, Yu Ying Chiu, Andy Lapastora, Peter Shen, Lexie Wang, Clevis Willrich, and Shane Steinert-Threlkeld. Filtered corpus training (fict) shows that language models can generalize from indirect evidence. *arXiv preprint arXiv:2405.15750*, 2024.

Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Leon Derczynski, Xingjian Du, Matteo Grella, Kranthi Gv, Xuzheng He, Haowen Hou, Przemyslaw Kazienko, Jan Kocon, Jiaming Kong, Bartłomiej Koptyra, Hayden Lau, Jiaju Lin, Krishna Sri Ipsit Mantri, Ferdinand Mom, Atsushi Saito, Guangyu Song, Xiangru Tang, Johan Wind, Stanisław Woźniak, Zhenyuan Zhang, Qinghua Zhou, Jian Zhu, and Rui-Jie Zhu. RWKV: Reinventing RNNs for the transformer era. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14048–14077, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.936. URL https://aclanthology.org/2023.findings-emnlp.936.

Laura Pérez-Mayos, Alba Táboas García, Simon Mille, and Leo Wanner. Assessing the syntactic capabilities of transformer-based multilingual language models. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3799–3812, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.333. URL https://aclanthology.org/2021.findings-acl.333.

Charles A Perfetti and Ying Liu. Orthography to phonology and meaning: Comparisons across and within writing systems. *Reading and Writing*, 18:193–210, 2005.

Matthew E. Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 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, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1179. URL https://aclanthology.org/D18-1179.

Aleksandar Petrov, Emanuele La Malfa, Philip Torr, and Adel Bibi. Language model tokenizers introduce unfairness between languages. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 36963–36990. Curran Associates, Inc., 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/74bb24dca8334adce292883b4b651eda-Paper-Conference.pdf.

Steven T Piantadosi. Modern language models refute chomsky's approach to language. *From fieldwork to linguistic theory: A tribute to Dan Everett*, pages 353–414, 2023.

Martin J. Pickering and Victor S. Ferreira. Structural priming: A critical review. *Psychological Bulletin*, 134:427–459, 2008. ISSN 1939-1455. doi: 10.1037/0033-2909.134.3.427.

Martin J. Pickering and Simon Garrod. Do people use language production to make predictions during comprehension? *Trends in Cognitive Sciences*, 11(3): 105–110, 2007. ISSN 1364-6613. doi: 10.1016/j.tics.2006.12.002. URL https://www.sciencedirect.com/science/article/pii/S1364661307000034.

Martin J. Pickering and Simon Garrod. An integrated theory of language production and comprehension. *Behavioral and Brain Sciences*, 36(4):329–347, 2013. ISSN 0140-525X, 1469-1825. doi: 10.1017/S0140525X12001495. URL https://www.cambridge.org/core/journals/behavioral-and-brain-sciences/article/an-integrated-theory-of-language-production-and-comprehension/B8078F8F7AAEE99DE0579ACF32039B6A.

Tiago Pimentel, Maria Ryskina, Sabrina J. Mielke, Shijie Wu, Eleanor Chodroff, Brian Leonard, Garrett Nicolai, Yustinus Ghanggo Ate, Salam Khalifa, Nizar Habash, Charbel El-Khaissi, Omer Goldman, Michael Gasser, William Lane, Matt Coler, Arturo Oncevay, Jaime Rafael Montoya Samame, Gema Celeste Silva Villegas, Adam Ek, Jean-Philippe Bernardy, Andrey Shcherbakov, Aziyana Bayyr-ool, Karina Sheifer, Sofya Ganieva, Matvey Plugaryov, Elena Klyachko, Ali Salehi, Andrew Krizhanovsky, Natalia Krizhanovsky, Clara Vania, Sardana Ivanova, Aelita Salchak, Christopher Straughn, Zoey Liu, Jonathan North Washington, Duygu Ataman, Witold Kieraś, Marcin Woliński, Totok Suhardijanto, Niklas Stoehr, Zahroh Nuriah, Shyam Ratan, Francis M. Tyers, Edoardo M. Ponti, Grant Aiton, Richard J. Hatcher, Emily Prud'hommeaux, Ritesh Kumar, Mans Hulden, Botond Barta, Dorina Lakatos, Gábor Szolnok, Judit Ács, Mohit Raj, David Yarowsky, Ryan Cotterell, Ben Ambridge, and Ekaterina Vylomova. SIGMORPHON 2021

shared task on morphological reinflection: Generalization across languages. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 229–259, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.sigmorphon-1.25. URL https://aclanthology.org/2021.sigmorphon-1.25.

Mārcis Pinnis, Rihards Krišlauks, Daiga Deksne, and Toms Miks. Neural machine translation for morphologically rich languages with improved sub-word units and synthetic data. In *Text, Speech, and Dialogue: 20th International Conference, TSD 2017, Prague, Czech Republic, August 27-31, 2017, Proceedings 20*, pages 237–245. Springer, 2017.

Frans Plank. Of abundance and scantiness in inflection: A typological prelude. *Paradigms: the economy of inflection*, pages 1–39, 1991.

Frans Plank. Split morphology: How agglutination and flexion mix. *Linguistic Typology*, 3:279–340, 1999.

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2362–2376, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.185. URL https://aclanthology.org/2020.emnlp-main.185.

Grusha Prasad, Marten van Schijndel, and Tal Linzen. Using Priming to Uncover the Organization of Syntactic Representations in Neural Language Models. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 66–76, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/K19-1007. URL https://aclanthology.org/K19-1007.

Prokopis Prokopidis and Haris Papageorgiou. Experiments for dependency parsing of Greek. In *Proceedings of the First Joint Workshop on Statistical Parsing of Morphologically Rich Languages and Syntactic Analysis of Non-Canonical Languages*, pages 90–96, Dublin, Ireland, August 2014. Dublin City University. URL https://aclanthology.org/W14-6109.

Ivan Provilkov, Dmitrii Emelianenko, and Elena Voita. BPE-dropout: Simple and effective subword regularization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1882–1892, Online, July 2020.

Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.170. URL https://aclanthology.org/2020.acl-main.170.

Jessie Quinn, Matthew Goldrick, Catherine Arnett, Victor S Ferreira, and Tamar H Gollan. Syntax drives default language selection in bilingual connected speech production. *Journal of experimental psychology. Learning, Memory, and Cognition*, 2024.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *OpenAI*, 2018. URL https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Technical Report*, 2019. URL https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, Eliza Rutherford, Tom Hennigan, Jacob Menick, Albin Cassirer, Richard Powell, George van den Driessche, Lisa Anne Hendricks, Maribeth Rauh, Po-Sen Huang, Amelia Glaese, Johannes Welbl, Sumanth Dathathri, Saffron Huang, Jonathan Uesato, John Mellor, Irina Higgins, Antonia Creswell, Nat McAleese, Amy Wu, Erich Elsen, Siddhant Jayakumar, Elena Buchatskaya, David Budden, Esme Sutherland, Karen Simonyan, Michela Paganini, Laurent Sifre, Lena Martens, Xiang Lorraine Li, Adhiguna Kuncoro, Aida Nematzadeh, Elena Gribovskaya, Domenic Donato, Angeliki Lazaridou, Arthur Mensch, Jean-Baptiste Lespiau, Maria Tsimpoukelli, Nikolai Grigorev, Doug Fritz, Thibault Sottiaux, Mantas Pajarskas, Toby Pohlen, Zhitao Gong, Daniel Toyama, Cyprien de Masson d'Autume, Yujia Li, Tayfun Terzi, Vladimir Mikulik, Igor Babuschkin, Aidan Clark, Diego de Las Casas, Aurelia Guy, Chris Jones, James Bradbury, Matthew Johnson, Blake Hechtman, Laura Weidinger, Iason Gabriel, William Isaac, Ed Lockhart, Simon Osindero, Laura Rimell, Chris Dyer, Oriol Vinyals, Kareem Ayoub, Jeff Stanway, Lorrayne Bennett, Demis Hassabis, Koray Kavukcuoglu, and Geoffrey Irving. Scaling Language Models: Methods, Analysis & Insights from Training Gopher, 2022. URL http://arxiv.org/abs/2112.11446.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer

learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

Alessandro Raganato and Jörg Tiedemann. An analysis of encoder representations in transformer-based machine translation. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 287–297, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5431. URL https://aclanthology.org/W18-5431.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1264. URL https://aclanthology.org/D16-1264.

Loganathan Ramasamy and Zdeněk Žabokrtský. Prague dependency style treebank for Tamil. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of Eighth International Conference on Language Resources and Evaluation (LREC 2012)*, pages 1888–1894, İstanbul, Turkey, 2012. ISBN 978-2-9517408-7-7. URL http://www.lrec-conf.org/proceedings/lrec2012/summaries/456.html.

Krithika Ramesh, Sunayana Sitaram, and Monojit Choudhury. Fairness in language models beyond English: Gaps and challenges. In Andreas Vlachos and Isabelle Augenstein, editors, *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2106–2119, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-eacl.157. URL https://aclanthology.org/2023.findings-eacl.157.

Surangika Ranathunga and Nisansa de Silva. Some languages are more equal than others: Probing deeper into the linguistic disparity in the NLP world. In Yulan He, Heng Ji, Sujian Li, Yang Liu, and Chua-Hui Chang, editors, *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 823–848, Online only, November 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.aacl-main.62.

Mohammad Sadegh Rasooli, Manouchehr Kouhestani, and Amirsaeid Moloodi. Development of a Persian syntactic dependency treebank. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 306–314, Atlanta, Georgia, June 2013. Association for Computational Linguistics. URL https://aclanthology.org/N13-1031.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. GPQA: A graduate-level google-proof q&a benchmark. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=Ti67584b98.

David Reitter, Frank Keller, and Johanna D. Moore. A Computational Cognitive Model of Syntactic Priming. *Cognitive Science*, 35(4):587–637, 2011. ISSN 1551-6709. doi: 10.1111/j.1551-6709.2010.01165.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1551-6709.2010.01165.x.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *2011 AAAI spring symposium series*, 2011.

Eiríkur Rögnvaldsson, Anton Karl Ingason, Einar Freyr Sigurðsson, and Joel Wallenberg. The Icelandic parsed historical corpus (IcePaHC). In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 1977–1984, Istanbul, Turkey, May 2012. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2012/pdf/440_Paper.pdf.

Joost Rommers and Kara D. Federmeier. Predictability's aftermath: Downstream consequences of word predictability as revealed by repetition effects. *Cortex*, 101: 16–30, 2018. ISSN 0010-9452. doi: 10.1016/j.cortex.2017.12.018. URL http://www.sciencedirect.com/science/article/pii/S0010945217304264.

Michael D. Rugg. The Effects of Semantic Priming and Word Repetition on Event-Related Potentials. *Psychophysiology*, 22(6):642–647, 1985. ISSN 1469-8986. doi: 10.1111/j.1469-8986.1985.tb01661.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1469-8986.1985.tb01661.x.

Michael D. Rugg. Event-related brain potentials dissociate repetition effects of high- and low-frequency words. *Memory & Cognition*, 18(4):367–379, 1990. ISSN 1532-5946. doi: 10.3758/BF03197126. URL https://doi.org/10.3758/BF03197126.

David E Rumelhart and James L McClelland. On learning the past tenses of english verbs. *Psycholinguistics: Critical Concepts in Psychology*, 4:216–271, 1986.

Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. How good is your tokenizer? on the monolingual performance of multilingual language models. 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 3118–3135, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.243. URL https://aclanthology.org/2021.acl-long.243.

Jonne Saleva and Constantine Lignos. The effectiveness of morphology-aware segmentation in low-resource neural machine translation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 164–174, Online, April 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-srw.22. URL https://aclanthology.org/2021.eacl-srw.22.

Dominiek Sandra. The morphology of the mental lexicon: Internal word structure viewed from a psycholinguistic perspective. *Language and cognitive processes*, 9 (3):227–269, 1994.

Naomi Saphra, Eve Fleisig, Kyunghyun Cho, and Adam Lopez. First tragedy, then parse: History repeats itself in the new era of large language models. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2310–2326, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.128. URL https://aclanthology.org/2024.naacl-long.128.

Michael Saxon, Ari Holtzman, Peter West, William Yang Wang, and Naomi Saphra. Benchmarks as microscopes: A call for model metrology. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=bttKwCZDkm.

Teven Le Scao, Angela Fan, Christopher Akiki, Elizabeth-Jane Pavlick, Suzana Ili'c, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, Franccois Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Rose Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas

Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, et al. Bloom: A 176b-parameter open-access multilingual language model. *arXiv*, 2022. URL https://arxiv.org/pdf/2211.05100.pdf.

Craig W Schmidt, Varshini Reddy, Haoran Zhang, Alec Alameddine, Omri Uzan, Yuval Pinter, and Chris Tanner. Tokenization is more than compression. *arXiv preprint arXiv:2402.18376*, 2024. URL https://arxiv.org/abs/2402.18376.

Sofie Schoonbaert, Robert J. Hartsuiker, and Martin J. Pickering. The representation of lexical and syntactic information in bilinguals: Evidence from syntactic priming. *Journal of Memory and Language*, 56(2):153–171, 2007. ISSN 0749-596X. doi: 10.1016/j.jml.2006.10.002. URL https://www.sciencedirect.com/science/article/pii/S0749596X06001471.

Abigail See, Aneesh Pappu, Rohun Saxena, Akhila Yerukola, and Christopher D. Manning. Do Massively Pretrained Language Models Make Better Storytellers? In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 843–861, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/K19-1079. URL https://aclanthology.org/K19-1079.

Amit Seker, Elron Bandel, Dan Bareket, Idan Brusilovsky, Refael Greenfeld, and Reut Tsarfaty. AlephBERT: Language Model Pre-training and Evaluation from Sub-Word to Sentence Level. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 46–56, Dublin, Ireland, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.4. URL https://aclanthology.org/2022.acl-long.4.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1162. URL https://aclanthology.org/P16-1162.

Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.

Hyopil Shin and Hyunjo You. Hybrid n-gram probability estimation in morphologically rich languages. In *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation*, pages 511–520. Waseda University, 2009.

Jeong-Ah Shin. Structural priming and L2 proficiency effects on bilingual syntactic processing in production. *Korean Journal of English Language and Linguistics*, 10 (3):499–518, 2010. doi: 10.15738/kjell.10.3.201009.499.

Jeong-Ah Shin and Kiel Christianson. Syntactic processing in Korean–English bilingual production: Evidence from cross-linguistic structural priming. *Cognition*, 112 (1):175–180, 2009. ISSN 0010-0277. doi: 10.1016/j.cognition.2009.03.011. URL `https://www.sciencedirect.com/science/article/pii/S0010027709000742`.

Anna Siewierska. Syntactic weight vs information structure and word order variation in Polish. *Journal of Linguistics*, 29(2):233–265, 1993. ISSN 1469-7742, 0022-2267. doi: 10.1017/S0022226700000323.

Kiril Simov, Petya Osenova, Alexander Simov, and Milen Kouylekov. Design and implementation of the bulgarian hpsg-based treebank. *Journal of Research on Language and Computation. Special Issue*, pages 495–522, 2005.

Arabella Sinclair, Jaap Jumelet, Willem Zuidema, and Raquel Fernández. Structural Persistence in Language Models: Priming as a Window into Abstract Language Representations. *Transactions of the Association for Computational Linguistics*, 10:1031–1050, 2022. ISSN 2307-387X. doi: 10.1162/tacl_a_00504. URL `https://doi.org/10.1162/tacl_a_00504`.

Shivalika Singh, Freddie Vargus, Daniel Dsouza, Börje F Karlsson, Abinaya Mahendiran, Wei-Yin Ko, Herumb Shandilya, Jay Patel, Deividas Mataciunas, Laura OMahony, et al. Aya dataset: An open-access collection for multilingual instruction tuning. *arXiv preprint arXiv:2402.06619*, 2024.

Koustuv Sinha, Jon Gauthier, Aaron Mueller, Kanishka Misra, Keren Fuentes, Roger Levy, and Adina Williams. Language model acceptability judgements are not always robust to context. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6043–6063, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.333. URL `https://aclanthology.org/2023.acl-long.333`.

Dan I Slobin. Cognitive prerequisites for the development of grammar. In *Studies of child language development*, pages 175–208. Holt, Rinehart, & Winston, 1973.

Dan I Slobin. Form-function relations: how do children find out what they are? *Language acquisition and conceptual development*, 3:406, 2001.

Dan I Slobin. Crosslinguistic evidence for the language-making capacity. In *The crosslinguistic study of language acquisition*, pages 1157–1256. Psychology Press, 2013.

Nathaniel Smith and Roger Levy. Cloze but no cigar: The complex relationship between cloze, corpus, and subjective probabilities in language processing. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 33, 2011.

Anders Søgaard. Should we ban English NLP for a year? In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5254–5260, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.351. URL https://aclanthology.org/2022.emnlp-main.351.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, et al. Dolma: An open corpus of three trillion tokens for language model pretraining research. *arXiv preprint arXiv:2402.00159*, 2024. URL https://arxiv.org/pdf/2402.00159.

Taiga Someya and Yohei Oseki. JBLiMP: Japanese benchmark of linguistic minimal pairs. In Andreas Vlachos and Isabelle Augenstein, editors, *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1581–1594, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-eacl.117. URL https://aclanthology.org/2023.findings-eacl.117.

Xinying Song, Alex Salcianu, Yang Song, Dave Dopson, and Denny Zhou. Fast WordPiece tokenization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2089–2103, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.160. URL https://aclanthology.org/2021.emnlp-main.160.

Yixiao Song, Kalpesh Krishna, Rajesh Bhatt, and Mohit Iyyer. SLING: Sino linguistic evaluation of large language models. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4606–4634, Abu Dhabi, United Arab Emirates,

December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.305. URL https://aclanthology.org/2022.emnlp-main.305.

Paul Soulos, Sudha Rao, Caitlin Smith, Eric Rosen, Asli Celikyilmaz, R. Thomas McCoy, Yichen Jiang, Coleman Haley, Roland Fernandez, Hamid Palangi, Jianfeng Gao, and Paul Smolensky. Structural Biases for Improving Transformers on Translation into Morphologically Rich Languages. In *Proceedings of the 4th Workshop on Technologies for MT of Low Resource Languages (LoResMT2021)*, pages 52–67, Virtual, 2021. Association for Machine Translation in the Americas. URL https://aclanthology.org/2021.mtsummit-loresmt.6.

Zayne Rea Sprague, Xi Ye, Kaj Bostrom, Swarat Chaudhuri, and Greg Durrett. MuSR: Testing the limits of chain-of-thought with multistep soft reasoning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=jenyYQzue1.

Adrian Staub, Margaret Grant, Lori Astheimer, and Andrew Cohen. The influence of cloze probability and item constraint on cloze task response time. *Journal of Memory and Language*, 82:1–17, 2015.

Patience Stevens and David C Plaut. From decomposition to distributed theories of morphological processing in reading. *Psychonomic Bulletin & Review*, 29(5): 1673–1702, 2022.

Michelle Suijkerbuijk, Zoë Prins, Marianne de Heer Kloots, Willem Zuidema, and Stefan L Frank. Blimp-nl: A corpus of dutch minimal pairsand acceptability judgements for languagemodel evaluation. *OSF Preprint*, 2024. URL https://osf.io/preprints/psyarxiv/mhjbx.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.824. URL https://aclanthology.org/2023.findings-acl.824.

Marcus Taft and Kenneth I Forster. Lexical storage and retrieval of polymorphemic and polysyllabic words. *Journal of verbal learning and verbal behavior*, 15(6): 607–620, 1976.

Tasmiah Tahsin Mayeesha, Abdullah Md Sarwar, and Rashedur M Rahman. Deep learning based question answering system in bengali. *Journal of Information and Telecommunication*, 5(2):145–178, 2021.

Samson Tan, Shafiq Joty, Lav Varshney, and Min-Yen Kan. Mind your inflections! Improving NLP for non-standard Englishes with Base-Inflection Encoding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5647–5663, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.455. URL https://aclanthology.org/2020.emnlp-main.455.

Hasan Tanvir, Claudia Kittask, Sandra Eiche, and Kairit Sirts. EstBERT: A pretrained language-specific BERT for Estonian. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 11–19, Reykjavik, Iceland (Online), May 31–2 June 2021. Linköping University Electronic Press, Sweden. URL https://aclanthology.org/2021.nodalida-main.2.

Yan Tao, Olga Viberg, Ryan S Baker, and René F Kizilcec. Cultural bias and cultural alignment of large language models. *PNAS nexus*, 3(9), 2024.

Mariona Taulé, M. Antònia Martí, and Marta Recasens. AnCora: Multilevel annotated corpora for Catalan and Spanish. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, May 2008. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2008/pdf/35_paper.pdf.

Ahmed Tawfik, Mahitab Emam, Khaled Essam, Robert Nabil, and Hany Hassan. Morphology-aware word-segmentation in dialectal Arabic adaptation of neural machine translation. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 11–17, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4602. URL https://aclanthology.org/W19-4602.

Wilson L Taylor. "cloze procedure": A new tool for measuring readability. *Journalism quarterly*, 30(4):415–433, 1953.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1452. URL https://aclanthology.org/P19-1452.

Together Computer. Redpajama: An open source recipe to reproduce llama training dataset, 2023. URL https://github.com/togethercomputer/RedPajama-Data.

Michael Tomasello. *Constructing a Language: A Usage-Based Theory of Language Acquisition*. Harvard University Press, 2003. ISBN 978-0-674-01030-7. doi: 10.2307/j.ctv26070v8. URL https://www.jstor.org/stable/j.ctv26070v8.

Cagri Toraman, Eyup Halit Yilmaz, Furkan Şahinuç, and Oguzhan Ozcelik. Impact of tokenization on language models: An analysis for turkish. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 22(4):1–21, 2023.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and Efficient Foundation Language Models, 2023a. URL http://arxiv.org/abs/2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b. URL https://arxiv.org/abs/2307.09288.

Sean Trott. Can large language models help augment english psycholinguistic datasets? *Behavior Research Methods*, pages 1–19, 2024a.

Sean Trott. Language models vs. llm-equipped software. *Substack*, 2024b. URL https://seantrott.substack.com/p/language-models-vs-llm-equipped-software. Accessed: 2024-11-18.

Daniela Trotta, Raffaele Guarasci, Elisa Leonardelli, and Sara Tonelli. Monolingual and cross-lingual acceptability judgments with the Italian CoLA corpus. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2929–2940, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-emnlp.250. URL https://aclanthology.org/2021.findings-emnlp.250.

Dimitra Irini Tzanidaki. Greek word order: towards a new approach. *UCL Working Paper in Linguistics*, 7:247–277, 1995. URL https://www.phon.ucl.ac.uk/publications/WPL/95papers/TZANIDAK.pdf.

Michael T Ullman. The declarative/procedural model: A neurobiological model of language learning, knowledge, and use. In *Neurobiology of language*, pages 953–968. Elsevier, 2016. URL https://www.sciencedirect.com/science/article/pii/B9780124077942000766.

Dennis Ulmer, Elisa Bassignana, Max Müller-Eberstein, Daniel Varab, Mike Zhang, Rob van der Goot, Christian Hardmeier, and Barbara Plank. Experimental standards for deep learning in natural language processing research. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2673–2692, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-emnlp.196. URL https://aclanthology.org/2022.findings-emnlp.196.

Unicode Consortium. The Unicode Standard, 2022. URL https://www.unicode.org/versions/Unicode15.0.0/UnicodeStandard-15.0.pdf.

Omri Uzan, Craig W Schmidt, Chris Tanner, and Yuval Pinter. Greed is All You Need: An Evaluation of Tokenizer Inference Methods. *arXiv preprint arXiv:2403.01289*, 2024. URL https://arxiv.org/abs/2403.01289.

Daan van Esch, Tamar Lucassen, Sebastian Ruder, Isaac Caswell, and Clara Rivera. Writing system and speaker metadata for 2,800+ language varieties. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5035–5046, Marseille, France, June 2022. European Language Resources Association. URL https://aclanthology.org/2022.lrec-1.538.

Roger P. G. van Gompel and Manabu Arai. Structural priming in bilinguals. *Bilingualism: Language and Cognition*, 21(3):448–455, 2018. ISSN 1366-7289, 1469-1841. doi: 10.1017/S1366728917000542. URL https://www.cambridge.org/core/journals/bilingualism-language-and-cognition/article/structural-priming-in-bilinguals/ABC53F9ABEAD3E7A0A6D0ECC456C8788.

Cyma Van Petten, Marta Kutas, Robert Kluender, Mark Mitchiner, and Heather McIsaac. Fractionating the Word Repetition Effect with Event-Related Potentials. *Journal of Cognitive Neuroscience*, 3(2):131–150, 1991. ISSN 0898-929X. doi: 10.1162/jocn.1991.3.2.131. URL https://doi.org/10.1162/jocn.1991.3.2.131.

Gaël Varoquaux, Alexandra Sasha Luccioni, and Meredith Whittaker. Hype, sustainability, and the price of the bigger-is-better paradigm in ai. *arXiv preprint arXiv:2409.14160*, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Tommi Vatanen, Jaakko J. Väyrynen, and Sami Virpioja. Language identification of short text segments with n-gram models. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, pages 3423–3430. European Language Resources Association (ELRA), 2010.

Veronika Vincze, Dóra Szauter, Attila Almási, György Móra, Zoltán Alexin, and János Csirik. Hungarian dependency treebank. In *Proceedings of the Seventh Conference on International Language Resources and Evaluation*, Valletta, Malta, 2010. URL `http://www.lrec-conf.org/proceedings/lrec2010/pdf/465_Paper.pdf`.

Elena Voita and Ivan Titov. Information-Theoretic Probing with Minimum Description Length. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 183–196, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.14. URL `https://aclanthology.org/2020.emnlp-main.14`.

Ekaterina Vylomova, Jennifer White, Elizabeth Salesky, Sabrina J. Mielke, Shijie Wu, Edoardo Maria Ponti, Rowan Hall Maudslay, Ran Zmigrod, Josef Valvoda, Svetlana Toldova, Francis Tyers, Elena Klyachko, Ilya Yegorov, Natalia Krizhanovsky, Paula Czarnowska, Irene Nikkarinen, Andrew Krizhanovsky, Tiago Pimentel, Lucas Torroba Hennigen, Christo Kirov, Garrett Nicolai, Adina Williams, Antonios Anastasopoulos, Hilaria Cruz, Eleanor Chodroff, Ryan Cotterell, Miikka Silfverberg, and Mans Hulden. SIGMORPHON 2020 shared task 0: Typologically diverse morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 1–39, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.sigmorphon-1.1. URL `https://aclanthology.org/2020.sigmorphon-1.1`.

Eric-Jan Wagenmakers and Simon Farrell. AIC model selection using Akaike weights. *Psychonomic bulletin & review*, 11:192–196, 2004. URL `https://link.springer.com/content/pdf/10.3758/BF03206482.pdf`.

Junxiong Wang, Tushaar Gangavarapu, Jing Nathan Yan, and Alexander M Rush. Mambabyte: Token-free selective state space model. In *First Conference on Language Modeling*, 2024a. URL `https://openreview.net/forum?id=X1xNsuKssb`.

Wentian Wang, Sarthak Jain, Paul Kantor, Jacob Feldman, Lazaros Gallos, and Hao Wang. MMLU-SR: A benchmark for stress-testing reasoning capability of large language models. In Dieuwke Hupkes, Verna Dankers, Khuyagbaatar Batsuren, Amirhossein Kazemnejad, Christos Christodoulopoulos, Mario Giulianelli, and Ryan Cotterell, editors, *Proceedings of the 2nd GenBench Workshop on Generalisation (Benchmarking) in NLP*, pages 69–85, Miami, Florida, USA, November 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.genbench-1.5. URL https://aclanthology.org/2024.genbench-1.5.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-pro: A more robust and challenging multi-task language understanding benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024c. URL https://openreview.net/forum?id=y10DM6R2r3.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641, 2019. doi: 10.1162/tacl_a_00290. URL https://aclanthology.org/Q19-1040.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. BLiMP: The Benchmark of Linguistic Minimal Pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392, 07 2020. ISSN 2307-387X. doi: 10.1162/tacl_a_00321. URL https://doi.org/10.1162/tacl_a_00321.

Alex Warstadt, Aaron Mueller, Leshem Choshen, Ethan Wilcox, Chengxu Zhuang, Juan Ciro, Rafael Mosquera, Bhargavi Paranjabe, Adina Williams, Tal Linzen, and Ryan Cotterell. Findings of the BabyLM challenge: Sample-efficient pretraining on developmentally plausible corpora. In Alex Warstadt, Aaron Mueller, Leshem Choshen, Ethan Wilcox, Chengxu Zhuang, Juan Ciro, Rafael Mosquera, Bhargavi Paranjabe, Adina Williams, Tal Linzen, and Ryan Cotterell, editors, *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, pages 1–34, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.conll-babylm.1. URL https://aclanthology.org/2023.conll-babylm.1.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc

V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Xiangpeng Wei, Haoran Wei, Huan Lin, Tianhao Li, Pei Zhang, Xingzhang Ren, Mei Li, Yu Wan, Zhiwei Cao, Binbin Xie, Tianxiang Hu, Shangjie Li, Binyuan Hui, Bowen Yu, Dayiheng Liu, Baosong Yang, Fei Huang, and Jun Xie. PolyLM: An Open Source Polyglot Large Language Model, 2023. URL `http://arxiv.org/abs/2307.06018`.

Leonie Weissweiler, Valentin Hofmann, Anjali Kantharuban, Anna Cai, Ritam Dutt, Amey Hengle, Anubha Kabra, Atharva Kulkarni, Abhishek Vijayakumar, Haofei Yu, Hinrich Schuetze, Kemal Oflazer, and David Mortensen. Counting the bugs in ChatGPT's wugs: A multilingual investigation into the morphological capabilities of a large language model. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6508–6524, Singapore, December 2023a. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.401. URL `https://aclanthology.org/2023.emnlp-main.401`.

Leonie Weissweiler, Valentin Hofmann, Abdullatif Köksal, and Hinrich Schütze. Explaining pretrained language models' understanding of linguistic structures using construction grammar. *Frontiers in Artificial Intelligence*, 6:1225791, 2023b.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL `https://aclanthology.org/2020.lrec-1.494`.

Ethan Wilcox, Roger Levy, and Richard Futrell. Hierarchical representation in neural language models: Suppression and recovery of expectations. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 181–190, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4819. URL `https://aclanthology.org/W19-4819`.

Ethan Wilcox, Pranali Vani, and Roger Levy. A targeted assessment of incremental processing in neural language models and humans. 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 939–952, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.76. URL https://aclanthology.org/2021.acl-long.76.

Clay Williams and Thomas Bever. Chinese character decoding: a semantic bias? *Reading and Writing*, 23:589–605, 2010.

Simon Willison. Think of language models like chatgpt as a "calculator for words", 2023. URL https://simonwillison.net/2023/Apr/2/calculator-for-words/. Accessed: 2024-10-20.

Genta Winata, Shijie Wu, Mayank Kulkarni, Thamar Solorio, and Daniel Preotiuc-Pietro. Cross-lingual Few-Shot Learning on Unseen Languages. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 777–791, Online only, 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.aacl-main.59.

Yonghui Wu, 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. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation, 2016. URL http://arxiv.org/abs/1609.08144.

Zhengxuan Wu, Isabel Papadimitriou, and Alex Tamkin. Oolong: Investigating What Makes Crosslingual Transfer Hard with Controlled Studies, 2022. URL http://arxiv.org/abs/2202.12312.

Mengzhou Xia, Mikel Artetxe, Chunting Zhou, Xi Victoria Lin, Ramakanth Pasunuru, Danqi Chen, Luke Zettlemoyer, and Veselin Stoyanov. Training trajectories of language models across scales. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13711–13738. Association for Computational Linguistics, 2023. URL https://aclanthology.org/2023.acl-long.767.

Beilei Xiang, Changbing Yang, Yu Li, Alex Warstadt, and Katharina Kann. CLiMP: A benchmark for Chinese language model evaluation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2784–2790, Online, April 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.eacl-main.242. URL https://aclanthology.org/2021.eacl-main.242.

Jin Xu, Xiaojiang Liu, Jianhao Yan, Deng Cai, Huayang Li, and Jian Li. Learning to Break the Loop: Analyzing and Mitigating Repetitions for Neural Text Generation. *Advances in Neural Information Processing Systems*, 35:3082–3095, 2022a. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/148c0aeea1c5da82f4fa86a09d4190da-Abstract-Conference.html.

Ningyu Xu, Tao Gui, Ruotian Ma, Qi Zhang, Jingting Ye, Menghan Zhang, and Xuanjing Huang. Cross-linguistic syntactic difference in multilingual BERT: How good is it and how does it affect transfer? In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8073–8092, Abu Dhabi, United Arab Emirates, December 2022b. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.552. URL https://aclanthology.org/2022.emnlp-main.552.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.41. URL https://aclanthology.org/2021.naacl-main.41.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. ByT5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306, 2022. doi: 10.1162/tacl_a_00461. URL https://aclanthology.org/2022.tacl-1.17.

Atsuki Yamaguchi, Aline Villavicencio, and Nikolaos Aletras. An Empirical Study on Cross-lingual Vocabulary Adaptation for Efficient Generative LLM Inference. *arXiv preprint arXiv:2402.10712*, 2024. URL https://arxiv.org/pdf/2402.10712.

Marat M. Yavrumyan and Anna S. Danielyan. Universal Dependencies and the Armenian Treebank. *Herald of the Social Sciences*, 2:231–244, 2020.

Shaked Yehezkel and Yuval Pinter. Incorporating context into subword vocabularies. In Andreas Vlachos and Isabelle Augenstein, editors, *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 623–635, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.eacl-main.45. URL https://aclanthology.org/2023.eacl-main.45.

Amir Zeldes. The GUM corpus: Creating multilayer resources in the classroom. *Language Resources and Evaluation*, 51(3):581–612, 2017. doi: http://dx.doi.org/10.1007/s10579-016-9343-x.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472.

Zhiyuan Zeng, Jiatong Yu, Tianyu Gao, Yu Meng, Tanya Goyal, and Danqi Chen. Evaluating large language models at evaluating instruction following. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=tr0KidwPLc.

Mengjiao Zhang and Jia Xu. Byte-based multilingual NMT for endangered languages. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4407–4417, Gyeongju, Republic of Korea, October 2022. International Committee on Computational Linguistics. URL https://aclanthology.org/2022.coling-1.388.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623, 2023.

Giulio Zhou. Morphological zero-shot neural machine translation, 2018.

Jayden Ziegler, Rodrigo Morato, and Jesse Snedeker. Priming semantic structure in Brazilian Portuguese. *Journal of Cultural Cognitive Science*, 3(1):25–37, 2019.

ISSN 2520-1018. doi: 10.1007/s41809-019-00022-8. URL https://doi.org/10.1007/s41809-019-00022-8.

Vilém Zouhar, Clara Meister, Juan Gastaldi, Li Du, Mrinmaya Sachan, and Ryan Cotterell. Tokenization and the noiseless channel. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5184–5207, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.284. URL https://aclanthology.org/2023.acl-long.284.

Appendix A

Chapter 2

A.1 NLLB Byte Premiums

Byte premiums calculated from NLLB are reported in Table A.1.

Table A.1: NLLB byte premiums. The byte premium for eng_latn is 1.0. Each language code is comprised of the ISO 639-3 (language) and ISO 15924 (script) codes separated by an underscore.

Language	**Byte premium**
ace_latn	1.2419926
afr_latn	1.0373004
aka_latn	1.5750612
als_latn	1.1673181
amh_ethi	1.7210862
arb_arab	1.4651134
asm_beng	2.5264323
ast_latn	1.7490516
awa_deva	2.7014324
ayr_latn	1.0976628
azb_arab	1.4901878
azj_latn	1.0761036
bak_cyrl	2.2716371
bam_latn	1.2569819
ban_latn	1.2695671
bem_latn	1.1553301
ben_beng	2.4308225
bho_deva	2.5153669
bod_tibt	2.6040539
bug_latn	1.2279017
bul_cyrl	1.8123562
cat_latn	1.0926706
ceb_latn	1.1134194
ces_latn	1.0358867
ckb_arab	1.6521034
ckb_arab	1.6521034
cym_latn	1.0265667
dan_latn	1.0211031
deu_latn	1.0537171
dik_latn	1.1239299

Table A.1: NLLB byte premiums. The byte premium for eng_latn is 1.0. Each language code is comprised of the ISO 639-3 (language) and ISO 15924 (script) codes separated by an underscore. (Continued)

Language	**Byte premium**
diq_latn	0.9590188
dyu_latn	1.1545521
dzo_tibt	3.2736977
ell_grek	1.9673049
ewe_latn	1.0783440
fao_latn	1.1557437
fij_latn	1.2107666
fin_latn	1.0589051
fon_latn	1.5413204
fra_latn	1.1742064
fur_latn	1.0672371
fuv_latn	1.1109194
gla_latn	0.9934613
gle_latn	1.9749562
glg_latn	1.0590246
guj_gujr	2.1627759
hau_latn	1.1766293
heb_hebr	1.3555346
hin_deva	2.3701629
hrv_latn	0.9897218
hun_latn	1.0199851
hye_armn	1.7241548
ibo_latn	1.3451287
ilo_latn	1.0765437
ind_latn	1.1788023
isl_latn	1.1543925
ita_latn	1.0669230
jav_latn	1.1468920
jpn_jpan	1.3220250
kab_latn	1.0287174
kac_latn	1.3451812

Table A.1: NLLB byte premiums. The byte premium for eng_latn is 1.0. Each language code is comprised of the ISO 639-3 (language) and ISO 15924 (script) codes separated by an underscore. (Continued)

Language	**Byte premium**
kam_latn	1.2177037
kan_knda	2.6420061
kas_arab	1.7762307
kas_deva	2.5259810
kat_geor	4.3381046
kbp_latn	1.4408085
kea_latn	0.7821679
khk_cyrl	1.8046135
khm_khmr	3.9051643
kik_latn	1.2930516
kin_latn	1.1340740
kir_cyrl	1.9635570
kmr_latn	1.0351712
knc_arab	2.5022926
knc_latn	1.1769876
kor_hang	1.2933602
lao_laoo	2.7071355
lij_latn	1.1438412
lin_latn	1.1393024
lit_latn	1.0300780
ltg_latn	1.0028570
ltz_latn	1.2253827
lug_latn	1.2175185
luo_latn	1.0358323
lus_latn	1.1689564
lvs_latn	1.2070388
mag_deva	2.5555142
mai_deva	2.3896953
mal_mlym	2.8852389
mar_deva	2.4793638
min_latn	0.9497956

Table A.1: NLLB byte premiums. The byte premium for eng_latn is 1.0. Each language code is comprised of the ISO 639-3 (language) and ISO 15924 (script) codes separated by an underscore. (Continued)

Language	**Byte premium**
mkd_cyrl	1.8349890
mlt_latn	1.0884567
mni_beng	3.0027416
mos_latn	1.1413713
mri_latn	1.1826053
mya_mymr	5.1034592
nld_latn	1.0516739
nob_latn	0.9977426
npi_deva	2.4202344
nus_latn	1.2935254
oci_latn	1.0146652
ory_orya	2.5109372
pag_latn	1.0439418
pan_guru	2.2208951
pbt_arab	1.7366557
pes_arab	1.5973940
plt_latn	1.1512264
pol_latn	1.0774161
por_latn	1.0979270
quy_latn	1.1639224
ron_latn	1.1151666
run_latn	1.1193204
rus_cyrl	1.8228284
sag_latn	1.1632489
san_deva	2.5428913
sat_beng	2.1131754
shn_mymr	2.8224643
sin_sinh	2.4463506
slk_latn	1.0415468
slv_latn	0.9722273
sna_latn	1.1192729
snd_arab	1.5880165

Table A.1: NLLB byte premiums. The byte premium for eng_latn is 1.0. Each language code is comprised of the ISO 639-3 (language) and ISO 15924 (script) codes separated by an underscore. (Continued)

Language	**Byte premium**
som_latn	1.4224149
sot_latn	1.1661078
spa_latn	1.0838621
srp_cyrl	1.4249495
sun_latn	1.0970417
swe_latn	1.0210256
swh_latn	1.0696621
tam_taml	2.7292892
taq_latn	1.2093634
tat_cyrl	1.8543562
tel_telu	2.6198705
tgk_cyrl	1.7469201
tgl_latn	1.1176348
tir_ethi	1.7631466
tuk_latn	1.7850561
tur_latn	1.0444815
tzm_tfng	1.9259158
uig_arab	2.3082357
ukr_cyrl	1.7514786
umb_latn	1.1673612
urd_arab	1.7079714
uzn_latn	1.6455453
vie_latn	1.3493725
wol_latn	1.0787309
xho_latn	1.1988860
ydd_hebr	1.8074376
yor_latn	1.3750599
zsm_latn	1.1438457
zul_latn	1.1639372

Appendix B

Chapter 4

B.1 MorphScore

The sources used for each language are:

- Bulgarian: UD_Bulgarian-BTB train split (Simov et al., 2005)
- English: UD_English-GUM train split (Zeldes, 2017)
- Spanish: UD_Spanish-AnCora train split (Taulé et al., 2008)
- Greek: UD_Greek-GUD train split (Prokopidis and Papageorgiou, 2014)
- Persian: UD_Persian-PerDT train split (Rasooli et al., 2013)
- Japanese: (Matsuzaki et al., 2024)
- Korean: UD_Korean-Kaist train split (Chun et al., 2018)
- Turkish: UniMorph (Pimentel et al., 2021)

- Indonesian: UD_Indonesian-GSD (Larasati et al., 2011)
- Hungarian: UD_Hungarian-Szeged train split (Vincze et al., 2010)
- Urdu: UD_Urdu-UDTB train split (Palmer et al., 2009; Bhat et al., 2017)
- Slovenian: UD_Slovenian-SSJ train split (Dobrovoljc et al., 2017; Dobrovoljc and Ljubešić, 2022)
- Tamil: UD_Tamil-TTB train split (Ramasamy and Žabokrtský, 2012)
- Georgian: UD_Georgian-GLC test split (Lobzhanidze)
- Armenian: UD_Armenian-BSUT train split (Yavrumyan and Danielyan, 2020)
- Irish: UD_Irish-IDT train split (Lynn and Foster, 2016)
- Icelandic: UD_Icelandic-Modern train split (Rögnvaldsson et al., 2012)
- Gujarati: UniMorph (Baxi and Bhatt, 2021)
- Kurdish: UniMorph (Kirov et al., 2018)
- Cebuano: UD_Cebuano-GJA test split (Aranes, Glyd Jun and Zeman, Dan , 2021)
- Basque: UD_Basque-BDT train split (Aranzabe et al., 2015)
- Zulu: UniMorph (Vylomova et al., 2020)

Appendix C

Chapter 5

C.1 Language Contamination in Multilingual Language Models

In this section, we estimate language contamination in CC-100-XL, the dataset used to train the XGLM models. While the dataset itself is not made available by Lin et al. (2022), the procedure used for language identification is similar to CC-100 (Conneau et al., 2020a; Wenzek et al., 2020).

While there are some differences in the approaches used for filtering languages to ensure high-quality data, both corpora are based on CommonCrawl snapshots and are divided into languages using the fastText language identification model (Joulin et al., 2017). Both CC-100 and CC-100-XL also involve a further language identification step. For CC-100, an unnamed internal tool is also used for language identification; for CC-100-XL, an additional step of language identification takes place where text

language is also identified at the paragraph level.

To test for Dutch and Polish contamination, we sample roughly 100M tokens (based on the XGLM 7.5B tokenizer) of all languages in the replicated CC-100 dataset[1] that XLGM 564M, 1.7B, 2.9B, and 7.5B are trained on. We only consider languages that have 100M or more tokens in CC-100 and that either use the Latin alphabet (Spanish, French, Italian, Portuguese, Finnish, Indonesian, Turkish, Vietnamese, Catalan, Estonian, Swahili, Basque), are Slavic (Russian, Bulgarian), or both (English, German). Specifically, we sample from each of these languages until we have enough documents that the number of tokens in each language is at least 100M. Thus, our sample of CC-100 includes roughly 1.6B tokens.

To replicate the additional filtering of CC-100-XL, we split all documents by paragraph and run language identification on them using the latest version of the fastText language identification model released as part of the "No Language Left Behind" project (Costa-jussà et al., 2022). We set the identification threshold to 0.5, which the authors find to be effective for lower-resource languages (which some of our sampled languages are among). We note that this is a newer and likely more accurate version of the language identification model than that used to create CC-100-XL, and thus it is even less likely to include data from languages other than those intended. We only analyze the data from paragraphs identified to be the same language as the document label.

To identify Dutch and Polish in these paragraphs, we divide paragraphs into sentences by splitting at each period character, and we run each sentence through

[1] `https://data.statmt.org/cc-100/`

both the aforementioned latest version of the fastText language identification model (Costa-jussà et al., 2022; Joulin et al., 2017) and the cld3 language identifier (Xue et al., 2021) as provided in the `gcld3` python package (Al-Rfou et al., 2020). We use a stricter threshold of 0.9 (as recommended for high-resource languages; Costa-jussà et al., 2022) for the former and use the default threshold of 0.7 for the latter.[2]

To estimate the total amount of contamination in each of these languages, we calculate the proportion of each language sample that includes Dutch or Polish. We then multiply this by the number of tokens in each language, which we estimate by multiplying the proportions given in Figure 1 of Lin et al. (2022) by 500B, the total number of tokens. We first provide two estimates of contamination for Dutch and Polish in Table C.1: the amount of contamination as identified by the fastText language identification model, and the amount identified by cld3. We also provide a third, more conservative estimate, that only includes the tokens that both language identification models identify as either Dutch or Polish. We note that because we only look at data from 16 of the 30 training languages, these numbers are likely to substantially underestimate the amount of language contamination in the XGLM pre-training data.

C.2 Statistical Tests

We provide the full results of the statistical tests for XGLM 4.5B (Table C.2), the PolyLMs (Table C.3), and the remaining XGLMs (Table C.4).

[2]See `https://github.com/google/cld3/blob/master/src/nnet_language_identifier.h` and `https://github.com/google/cld3/blob/master/src/nnet_language_identifier.cc`.

Table C.1: Estimated Dutch and Polish contamination in the training data of XGLM 564M, 1.7B, 2.9B, and 7.5B, based on language identification using cld3 and fastText, only considering tokens that both language identification models predict to be Dutch or Polish.

	Dutch		**Polish**	
Language ID Tool	**Proportion**	**Estimated Tokens**	**Proportion**	**Estimated Tokens**
cld3	0.03051%	152,528,079	0.00668%	33,418,112
fastText	0.00212%	10,595,403	0.00157%	7,841,824
Consensus	0.00035%	1,774,765	0.00029%	1,456,856
(cld3 + fastText)				

Table C.2: Statistical tests of structural priming for XGLM 4.5B.

Language Model	**Language Pair**	**F**	**df_1**	**df_2**	**p**
XGLM 4.5B	Dutch→English	151.98	1	144	<0.0001
	English→Dutch	24.00	1	141	<0.0001
	Mandarin→Mandarin	192.37	1	24	<0.0001
	Mandarin→Mandarin	419.66	1	32	<0.0001
	English→Polish	1.35	1	31	0.2955
	Polish→English	0.96	1	32	0.3704
	English→Spanish	9.17	1	112	0.0056
	Spanish→English	4.33	1	112	0.0558
	English→Greek	7.28	1	24	0.0201
	Greek→English	5.05	1	24	0.0485
	Greek→Greek	8.40	1	24	0.0132
	English→German	0.13	1	16	0.7462
	German→English	0.10	1	16	0.7647
	Dutch→Dutch	385.71	1	144	<0.0001
	Dutch→English	57.28	1	144	<0.0001
	English→Dutch	134.53	1	137	<0.0001

Table C.3: Statistical tests of structural priming for PolyLM 1.7B and 13B.

Language Model	**Language Pair**	**F**	**df$_1$**	**df$_2$**	**p**
PolyLM 1.7B	Dutch→English	116.87	1	144	<0.0001
	English→Dutch	18.80	1	144	<0.0001
	Mandarin→Mandarin	164.45	1	24	<0.0001
	Mandarin→Mandarin	228.25	1	32	<0.0001
	English→Polish	7.50	1	32	0.0165
	Polish→English	7.34	1	32	0.0174
	English→Spanish	2.47	1	112	0.1498
	Spanish→English	1.76	1	112	0.2280
	English→Greek	0.13	1	24	0.7462
	Greek→English	0.13	1	24	0.7462
	Greek→Greek	8.50	1	24	0.0128
	English→German	1.39	1	16	0.2955
	German→English	2.66	1	16	0.1525
	Dutch→Dutch	105.51	1	144	<0.0001
	Dutch→English	55.84	1	144	<0.0001
	English→Dutch	140.97	1	144	<0.0001
PolyLM 13B	Dutch→English	193.43	1	144	<0.0001
	English→Dutch	16.73	1	144	0.0002
	Mandarin→Mandarin	141.67	1	24	<0.0001
	Mandarin→Mandarin	257.28	1	32	<0.0001
	English→Polish	2.45	1	32	0.1570
	Polish→English	0.29	1	32	0.6275
	English→Spanish	21.87	1	112	<0.0001
	Spanish→English	41.60	1	112	<0.0001
	English→Greek	0.70	1	24	0.4481
	Greek→English	0.54	1	24	0.5062
	Greek→Greek	9.03	1	24	0.0106
	English→German	5.36	1	16	0.0485
	German→English	1.51	1	16	0.2794
	Dutch→Dutch	260.25	1	144	<0.0001
	Dutch→English	129.76	1	144	<0.0001
	English→Dutch	58.52	1	144	<0.0001

Table C.4: Statistical tests of structural priming for XGLM 564M, 1.7B, 2.9B, and 7.5B.

Language Model	Study	Language Pair	F	df_1	df_2	p
XGLM 564M	Dutch→English	12.89	1	144	0.0010	
	English→Dutch	16.59	1	144	0.0002	
	Mandarin→Mandarin	301.39	1	24	<0.0001	
	Mandarin→Mandarin	1006.36	1	32	<0.0001	
	English→Polish	1.05	1	32	0.3497	
	Polish→English	10.30	1	32	0.0056	
	English→Spanish	0.51	1	112	0.5076	
	Spanish→English	4.72	1	112	0.0471	
	English→Greek	5.90	1	24	0.0352	
	Greek→English	11.25	1	24	0.0051	
	Greek→Greek	10.80	1	24	0.0056	
	English→German	3.65	1	16	0.1001	
	German→English	2.76	1	16	0.1494	
	Dutch→Dutch	545.14	1	144	<0.0001	
	Dutch→English	5.66	1	144	0.0291	
	English→Dutch	55.69	1	144	<0.0001	
XGLM 1.7B	Dutch→English	17.64	1	144	0.0001	
	English→Dutch	32.57	1	144	<0.0001	
	Mandarin→Mandarin	751.15	1	24	<0.0001	
	Mandarin→Mandarin	1519.71	1	32	<0.0001	
	English→Polish	0.08	1	32	0.7761	
	Polish→English	0.69	1	32	0.4481	
	English→Spanish	4.76	1	112	0.0467	
	Spanish→English	3.19	1	112	0.1026	
	English→Greek	2.62	1	24	0.1502	
	Greek→English	11.20	1	24	0.0051	
	Greek→Greek	18.49	1	24	0.0005	
	English→German	1.80	1	16	0.2358	
	German→English	3.13	1	16	0.1247	
	Dutch→Dutch	312.38	1	144	<0.0001	
	Dutch→English	3.72	1	144	0.0770	
	English→Dutch	55.88	1	134	<0.0001	

Table C.4: Statistical tests of structural priming for XGLM 564M, 1.7B, 2.9B, and 7.5B. (Continued)

Language Model	Study	Language Pair	F	df_1	df_2	p
XGLM 2.9B	Dutch→English	47.12	1	144	<0.0001	
	English→Dutch	27.25	1	144	<0.0001	
	Mandarin→Mandarin	427.12	1	24	<0.0001	
	Mandarin→Mandarin	1363.62	1	32	<0.0001	
	English→Polish	1.31	1	32	0.2988	
	Polish→English	12.11	1	32	0.0031	
	English→Spanish	4.61	1	112	0.0489	
	Spanish→English	10.42	1	112	0.0033	
	English→Greek	3.58	1	24	0.0966	
	Greek→English	12.26	1	24	0.0036	
	Greek→Greek	16.05	1	24	0.0011	
	English→German	6.22	1	16	0.0362	
	German→English	1.11	1	16	0.3485	
	Dutch→Dutch	327.66	1	144	<0.0001	
	Dutch→English	21.01	1	144	<0.0001	
	English→Dutch	90.89	1	144	<0.0001	
XGLM 7.5B	Dutch→English	37.88	1	144	<0.0001	
	English→Dutch	21.46	1	144	<0.0001	
	Mandarin→Mandarin	402.46	1	24	<0.0001	
	Mandarin→Mandarin	1193.10	1	32	<0.0001	
	English→Polish	0.08	1	32	0.7761	
	Polish→English	8.96	1	32	0.0093	
	English→Spanish	16.41	1	112	0.0002	
	Spanish→English	17.28	1	112	0.0002	
	English→Greek	3.10	1	24	0.1202	
	Greek→English	12.33	1	24	0.0036	
	Greek→Greek	9.47	1	24	0.0092	
	English→German	1.86	1	16	0.2310	
	German→English	6.84	1	16	0.0291	
	Dutch→Dutch	402.81	1	144	<0.0001	
	Dutch→English	43.84	1	144	<0.0001	
	English→Dutch	83.09	1	144	<0.0001	

Appendix D

Chapter 6

D.1 Introduction

D.2 Model Training Details

Model training code is based on that from Chang and Bergen (2022)[1]. In total, model training took approximately 512 GPU hours on one NVIDIA RTX A6000. The estimated carbon emission for training all models was 66 kg CO_2 eq.[2].

Model Hyperparameters

[1] Available at `https://github.com/tylerachang/word-acquisition-language-models`
[2] Carbon emissions were calculated via `https://mlco2.github.io/impact/#compute`

Table D.1: Language model hyperparameters

Hyperparameter	**Value**
Layers	12
Embedding size	768
Hidden size	768
Intermediate hidden size	3072
Attention heads	12
Attention head size	64
Activation function	GELU
Vocab size	50004
Max sequence length	128
Position embedding	Absolute
Batch size	256
Train steps	1M
Learning rate decay	Linear
Warmup steps	10000
Learning rate	1e-4
Adam ϵ	1e-6
Adam β_1	0.9
Adam β_2	0.999
Dropout	0.1
Attention dropout	0.1

Checkpoints We take checkpoints at the first and last steps (128k). Additionally we take checkpoints every 10k steps. After the introduction of the L2 at the halfway point (64k), we save checkpoints every 10 steps, because we expect that structural priming effects may emerge within the first few hundred training steps after the introduction of L2. After 200 steps after the introduction of L2, we gradually increase the checkpoint intervals. This way, we have increased resolution during the period of training where we expect to see the emergence of structural priming effects, while

minimizing the number of checkpoints needed.

We save model checkpoints at the following training steps: 0, 10000, 20000, 30000, 40000, 50000, 64000, 64010, 64020, 64030, 64040, 64050, 64060, 64070, 64080, 64090, 64100, 64110, 64120, 64130, 64140, 64150, 64160, 64170, 64180, 64190, 64200, 64300, 64400, 64500, 64600, 64700, 64800, 64900, 65000, 66000, 67000, 68000, 69000, 70000, 80000, 90000, 100000, 110000, 120000, 128000.

D.3 L2-L1 Priming

Figures D.1 and D.2 show the L2→L1 for all models for both the simultaneous and sequential bilingual conditions, respectively. Each facet represents a model. The labels, e.g. English-Dutch and Dutch-English correspond to the L1 and L2 of each model.

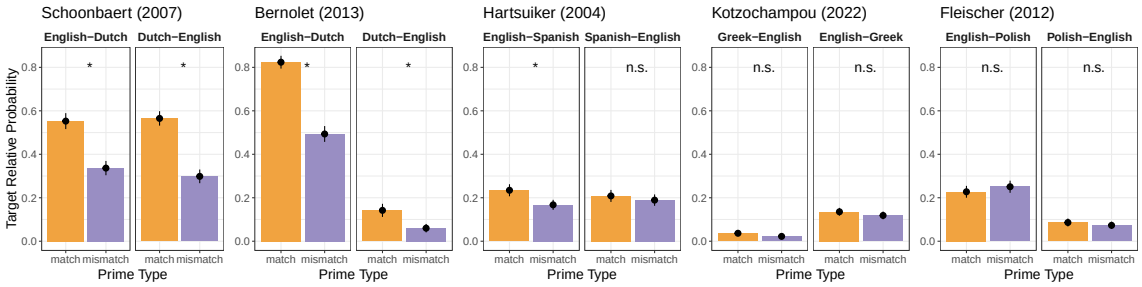

Figure D.1: Simultaneous bilingual condition. Prime language corresponds to L2.

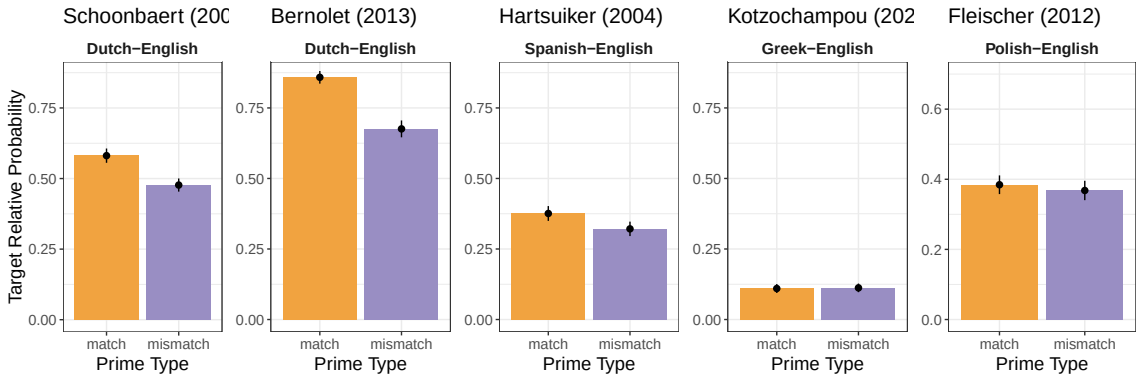

Figure D.2: Sequential bilingual condition. Prime language corresponds to L2.

D.4 All Training Dynamics Results

D.4.1 Schoonbaert (2007)

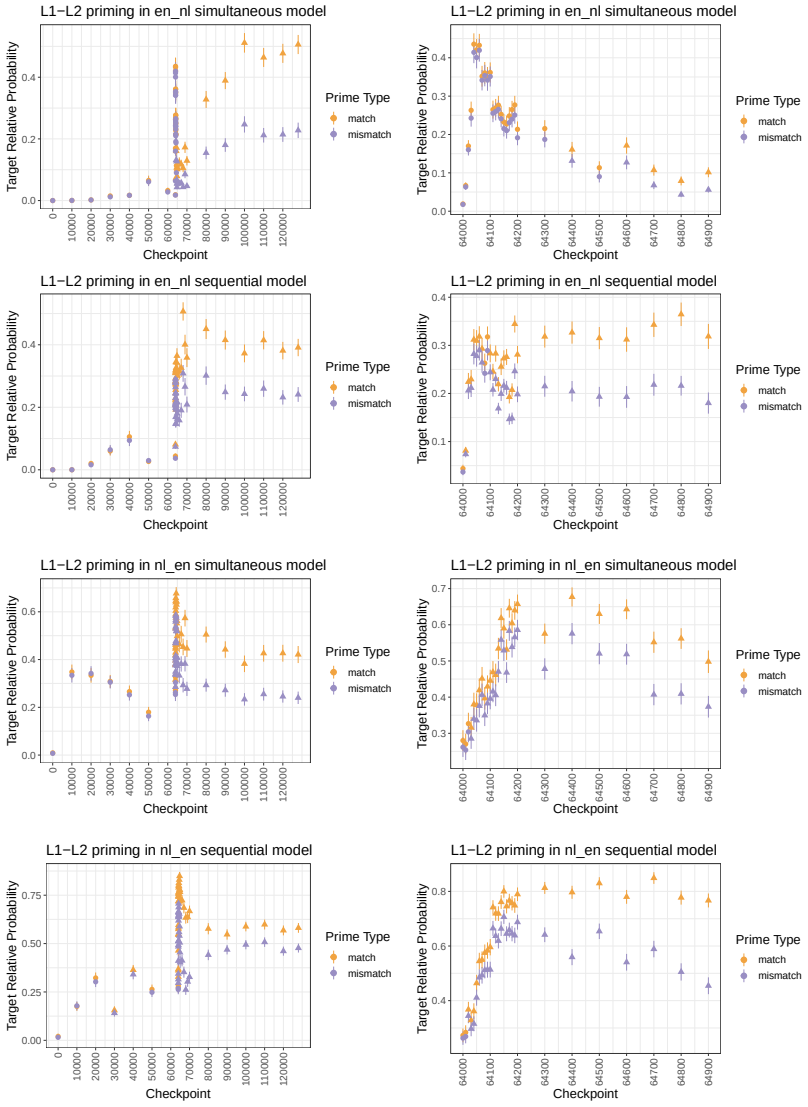

Figure D.3: L1-L2 structural priming effects over the course of training for Dutch and English models with the Schoonbaert et al. (2007) stimuli.

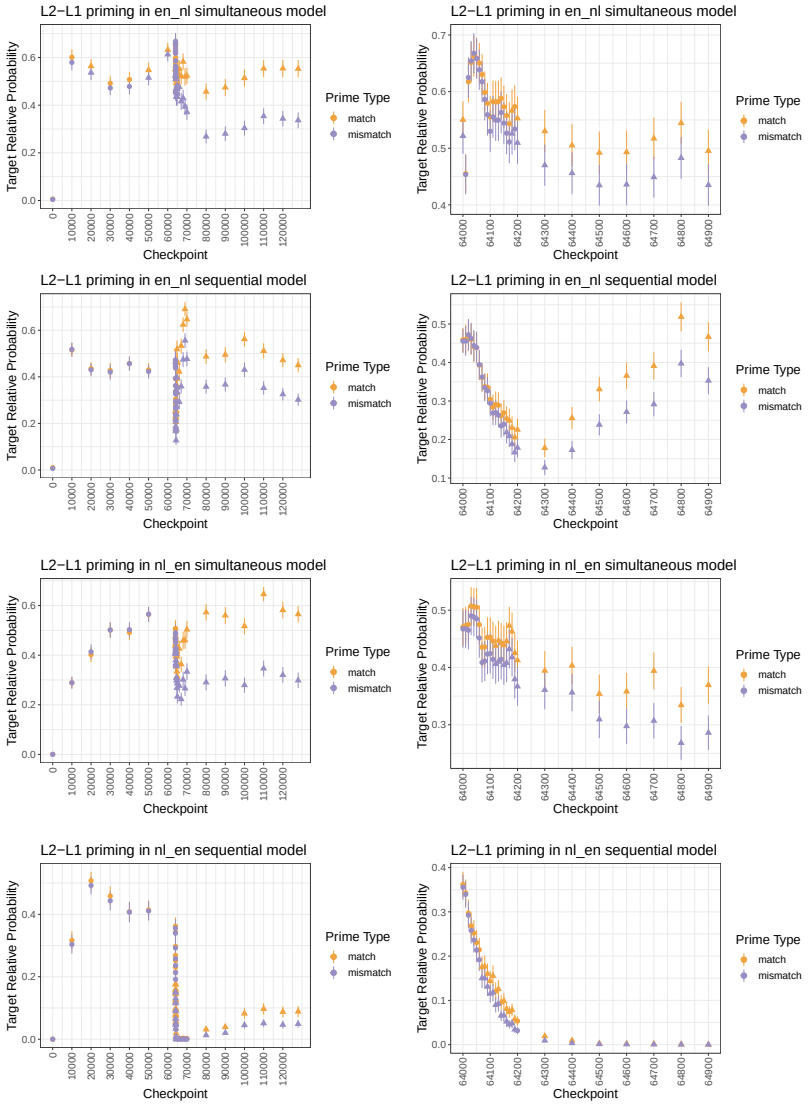

Figure D.4: L2-L1 structural priming effects over the course of training for Dutch and English models with the Schoonbaert et al. (2007) stimuli.

D.4.2 Bernolet (2013)

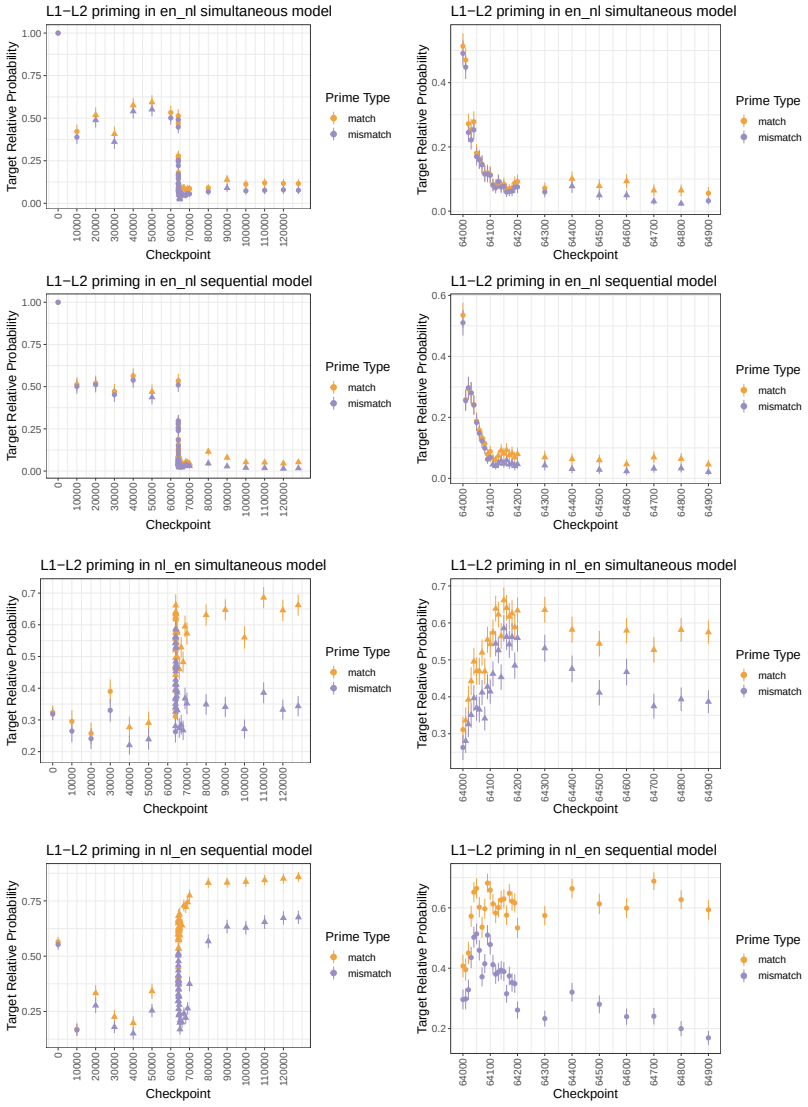

Figure D.5: L1-L2 structural priming effects over the course of training for Dutch and English models with the Bernolet et al. (2013) stimuli.

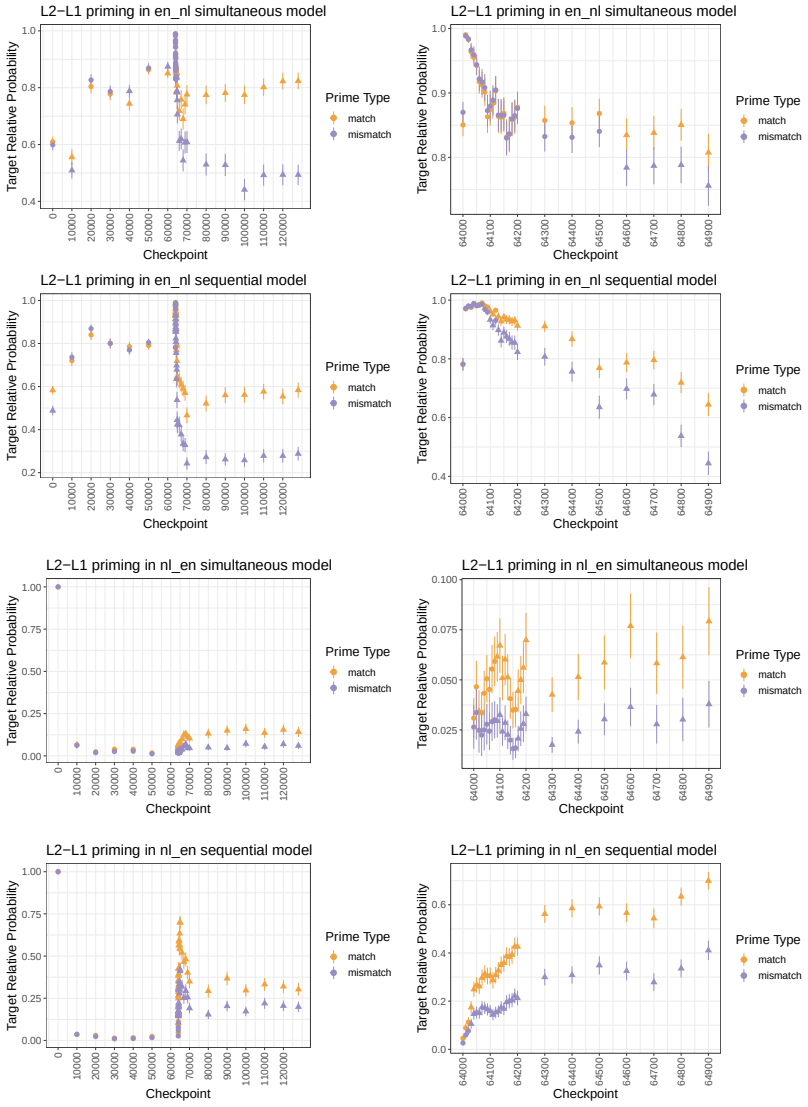

Figure D.6: L2-L1 structural priming effects over the course of training for Dutch and English models with the Bernolet et al. (2013) stimuli.

D.4.3 Hartsuiker (2004)

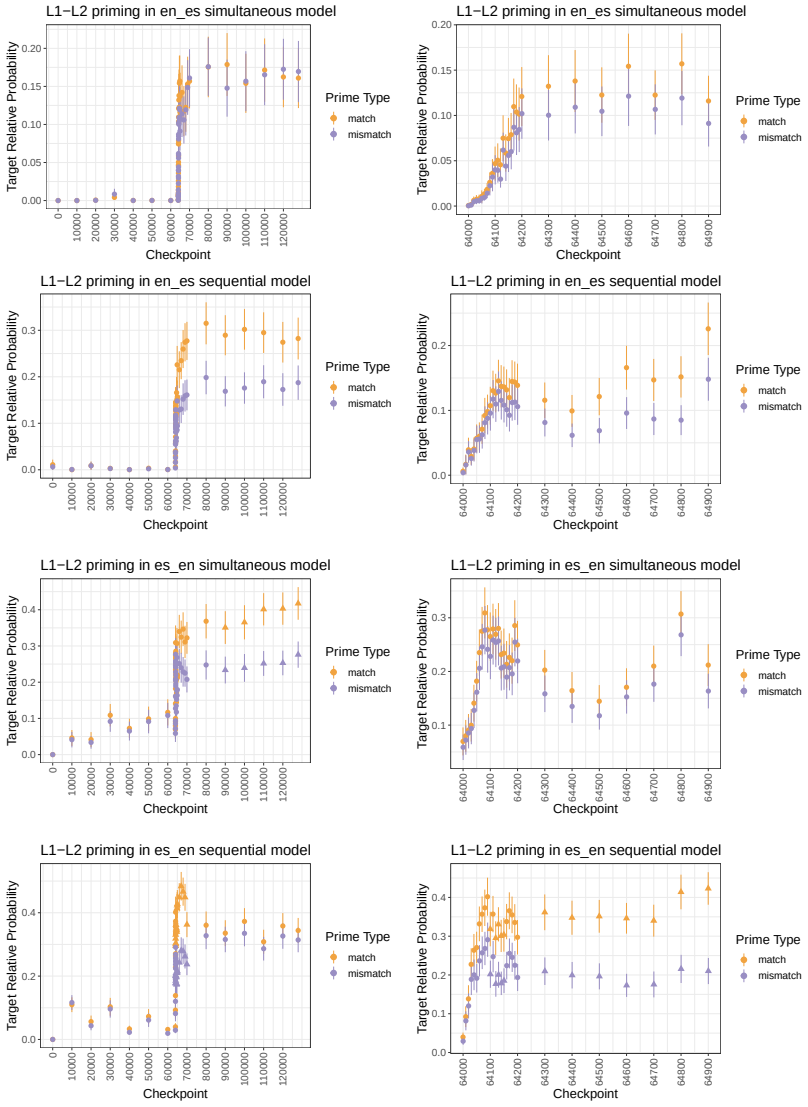

Figure D.7: L1-L2 structural priming effects over the course of training for Spanish and English models with the Hartsuiker et al. (2004) stimuli.

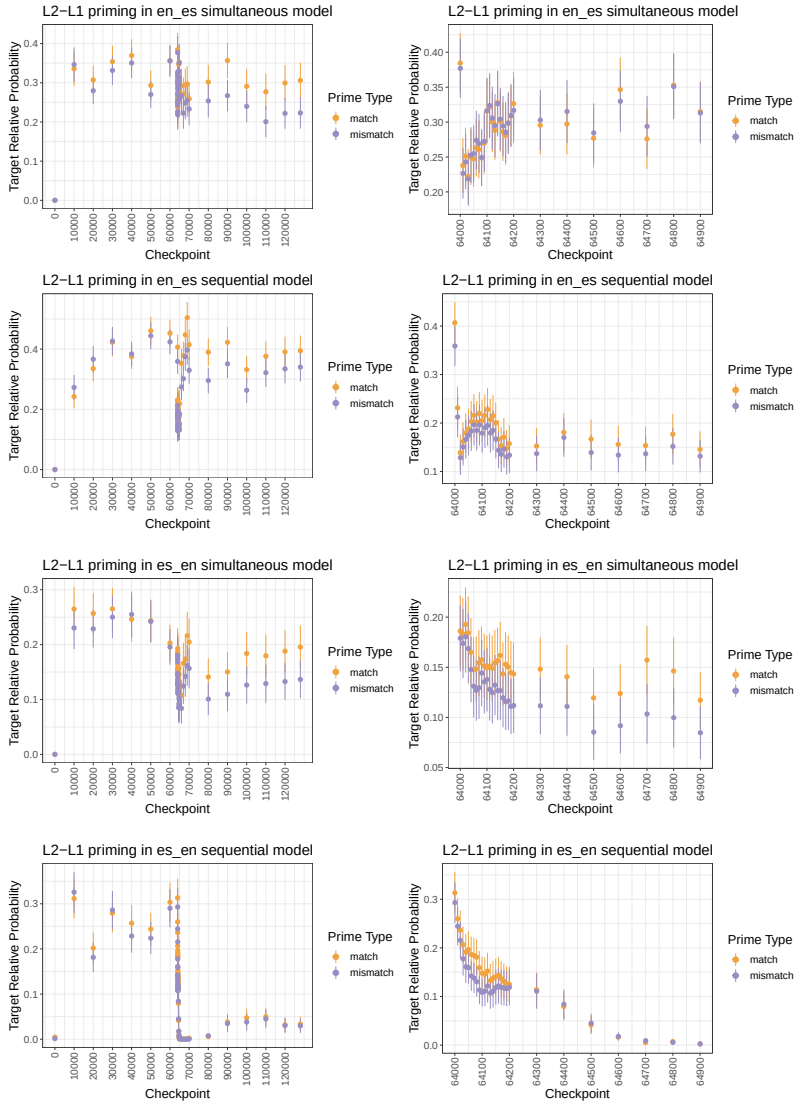

Figure D.8: L2-L1 structural priming effects over the course of training for Spanish and English models with the Hartsuiker et al. (2004) stimuli.

D.4.4 Fleischer (2012)

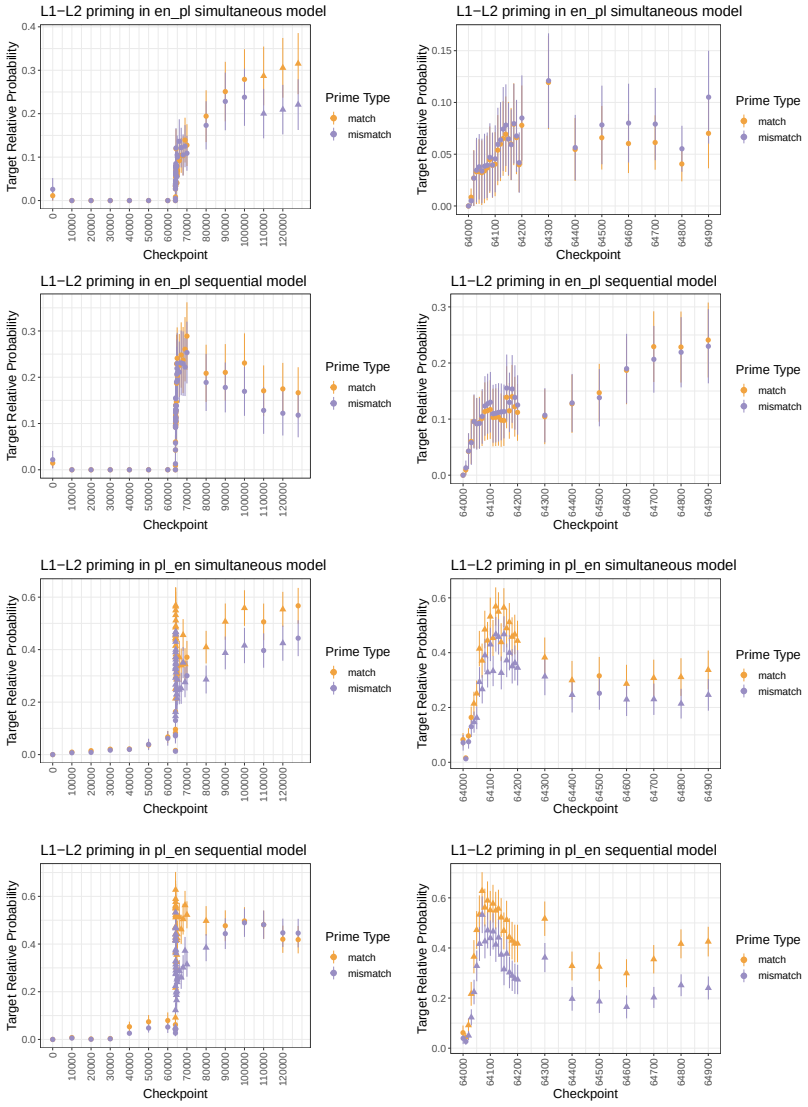

Figure D.9: L1-L2 structural priming effects over the course of training for Polish and English models with the Fleischer et al. (2012) stimuli.

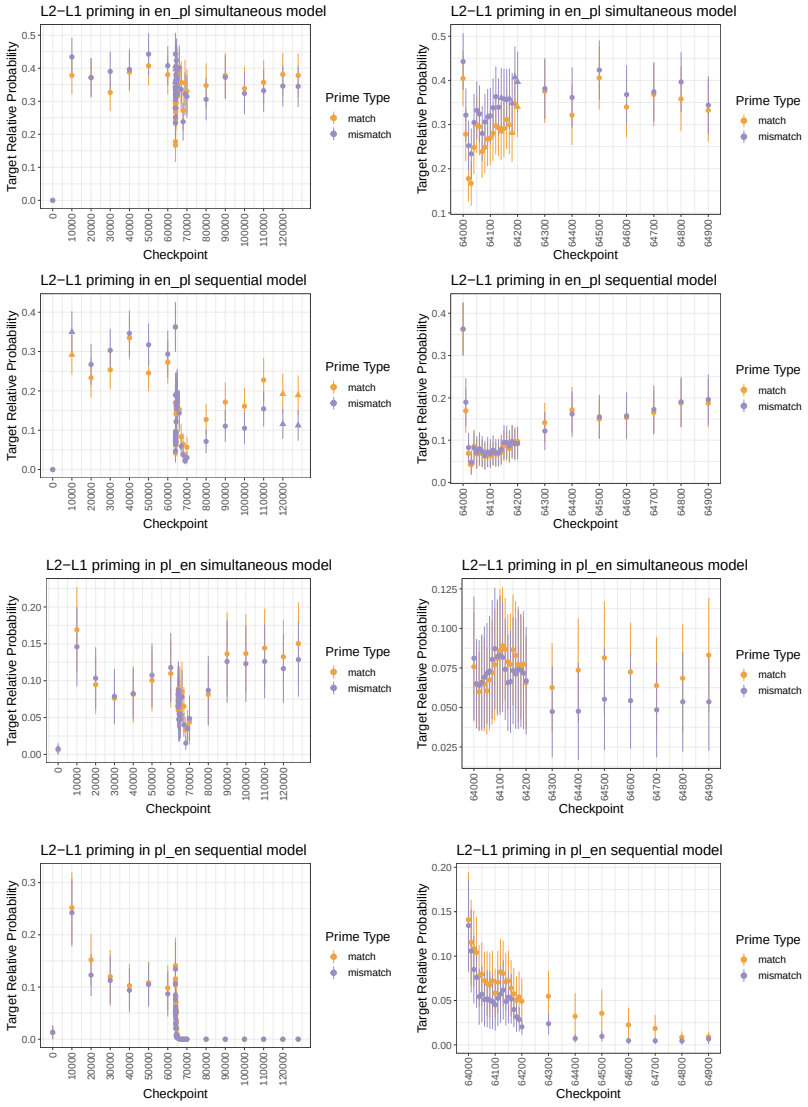

Figure D.10: L2-L1 structural priming effects over the course of training for Polish and English models with the Fleischer et al. (2012) stimuli.

D.4.5 Kotzochampou (2022)

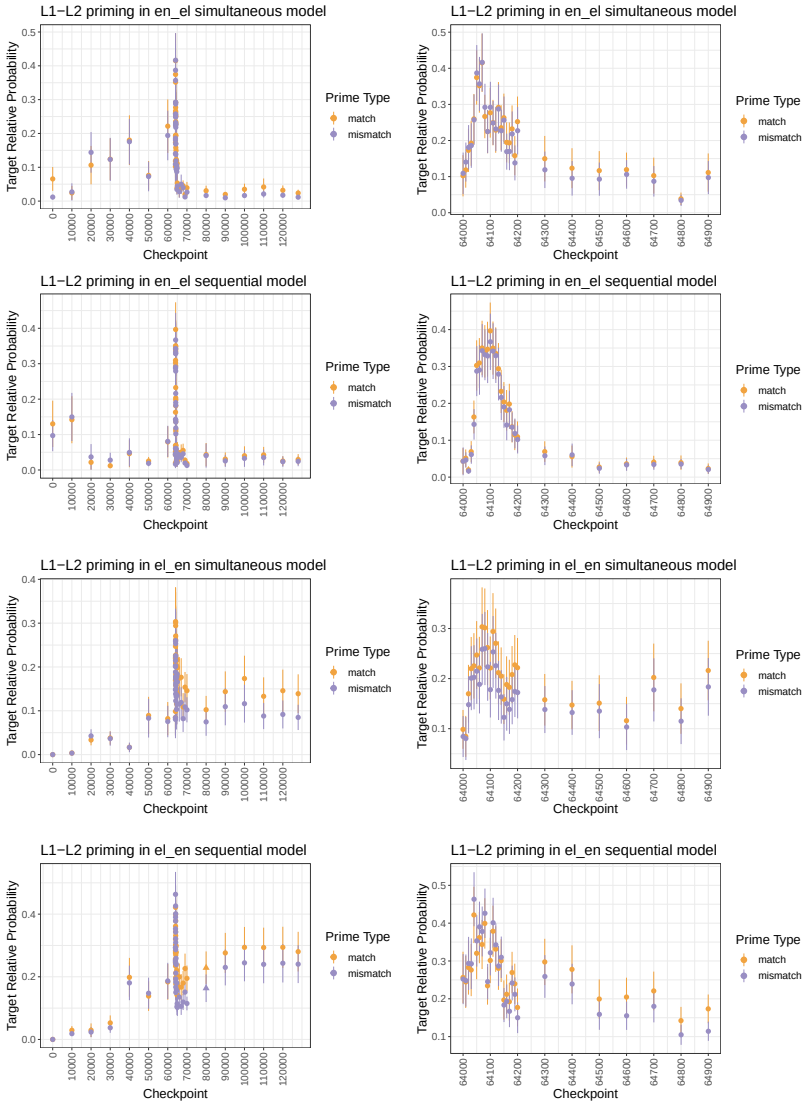

Figure D.11: L1-L2 structural priming effects over the course of training for Greek and English models with the Kotzochampou and Chondrogianni (2022) stimuli.

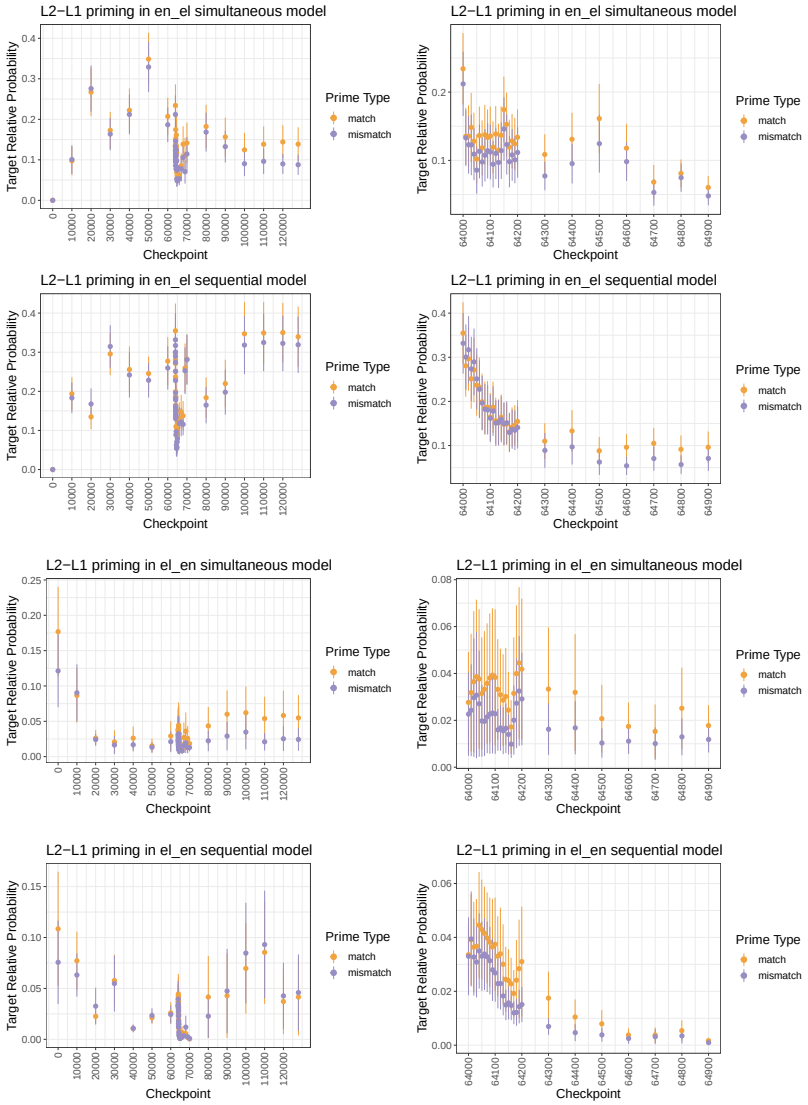

Figure D.12: L2-L1 structural priming effects over the course of training for Greek and English models with the Kotzochampou and Chondrogianni (2022) stimuli.

D.5 Structural Priming and Loss

D.5.1 Schoonbaert (2007)

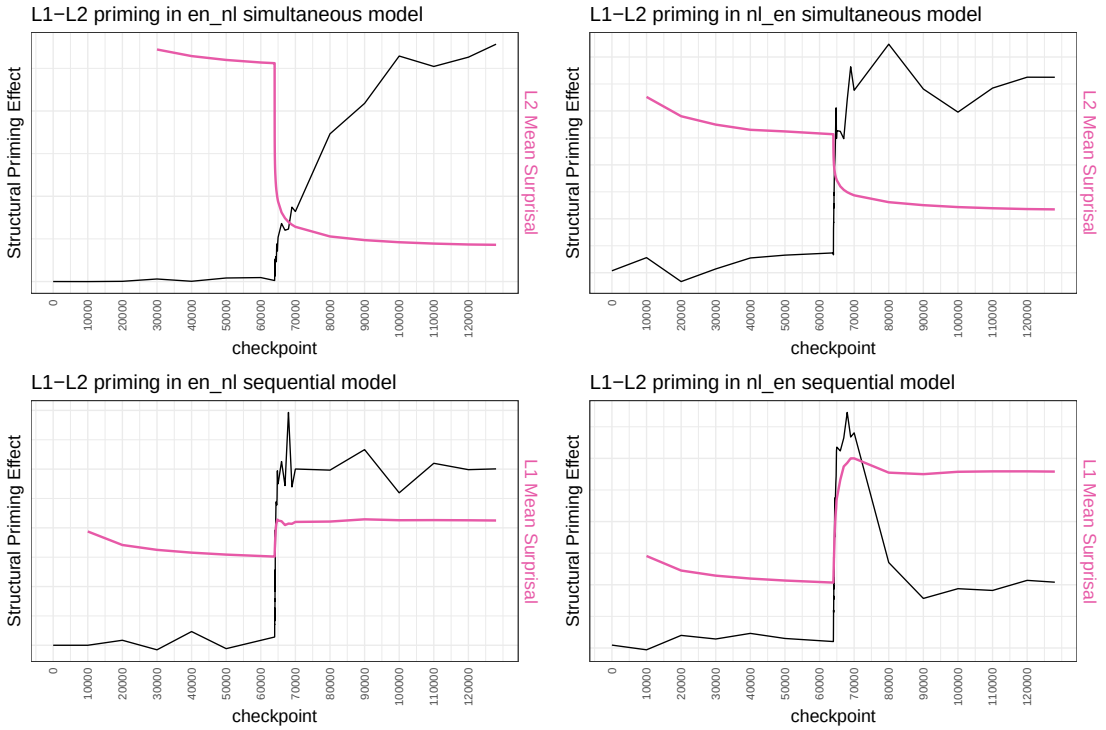

Figure D.13: Structural priming effect (black), plotted as the difference between match and mismatch conditions, and mean surprisal (pink) over the course of model training. Y-axes have been re-scaled for easier comparison.

D.5.2 Bernolet (2013)

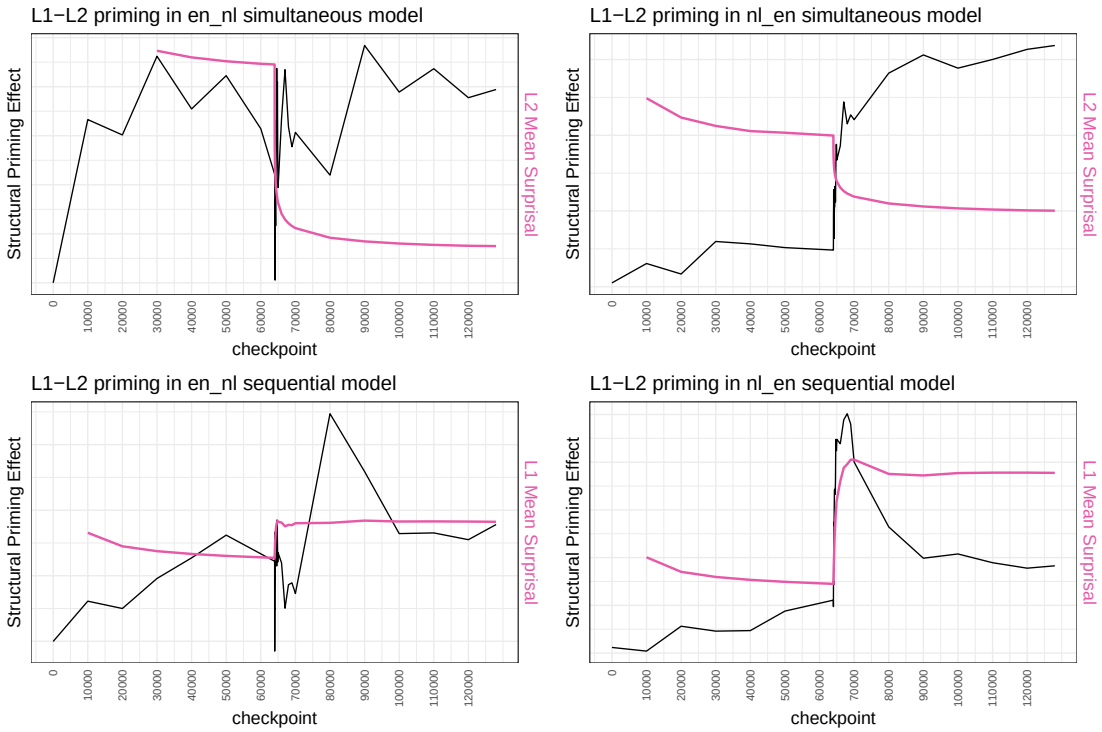

Figure D.14: Structural priming effect (black), plotted as the difference between match and mismatch conditions, and mean surprisal (pink) over the course of model training. Y-axes have been re-scaled for easier comparison.

D.5.3 Hartsuiker (2004)

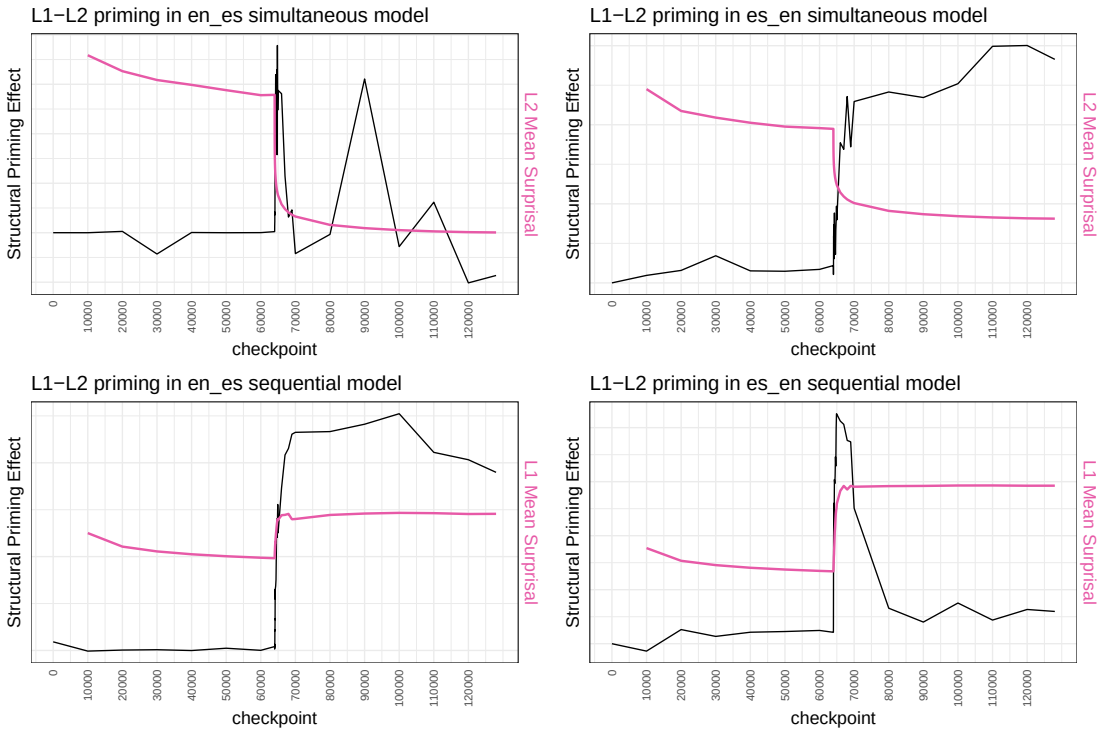

Figure D.15: Structural priming effect (black), plotted as the difference between match and mismatch conditions, and mean surprisal (pink) over the course of model training. Y-axes have been re-scaled for easier comparison.

D.5.4 Fleischer (2012)

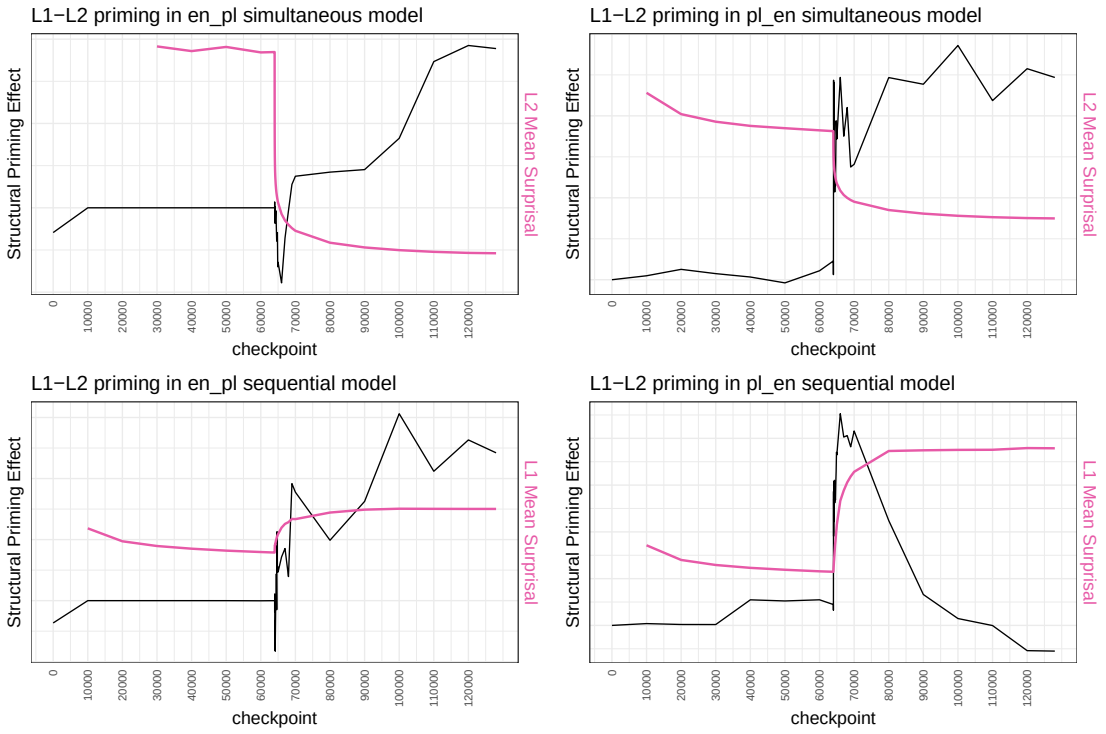

Figure D.16: Structural priming effect (black), plotted as the difference between match and mismatch conditions, and mean surprisal (pink) over the course of model training. Y-axes have been re-scaled for easier comparison.

D.5.5 Kotzochampou (2022)

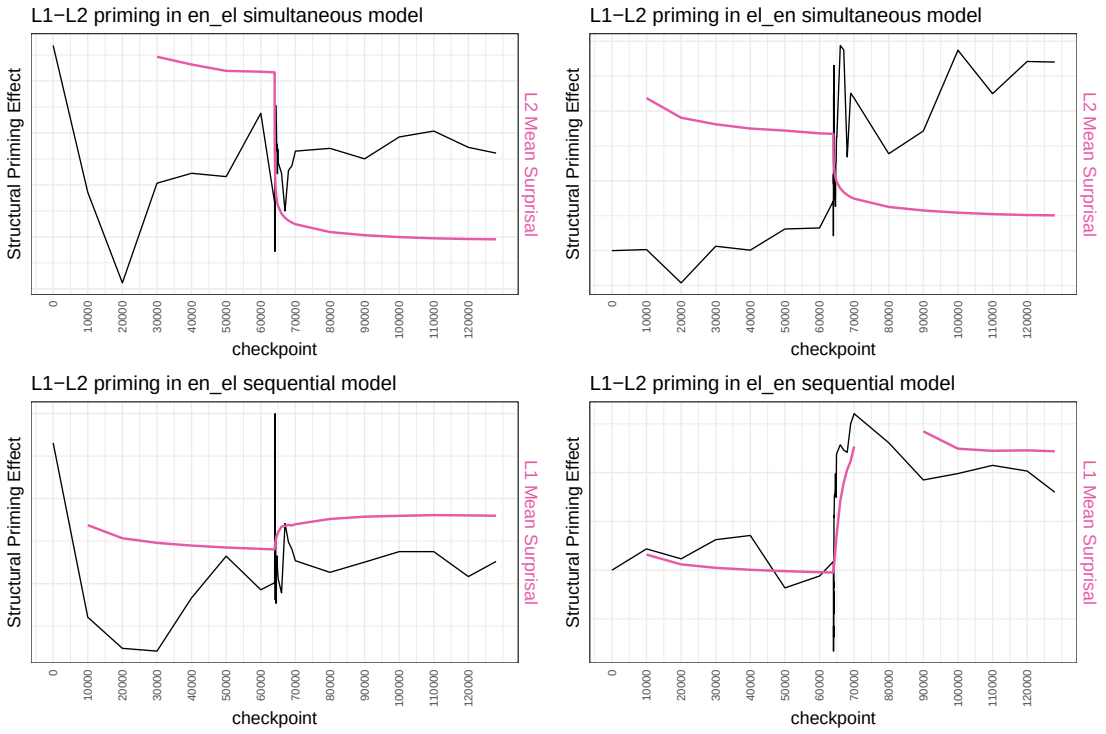

Figure D.17: Structural priming effect (black), plotted as the difference between match and mismatch conditions, and mean surprisal (pink) over the course of model training. Y-axes have been re-scaled for easier comparison.

D.6 BLiMP and SP Training Dynamics

D.6.1 Schoonbaert (2007)

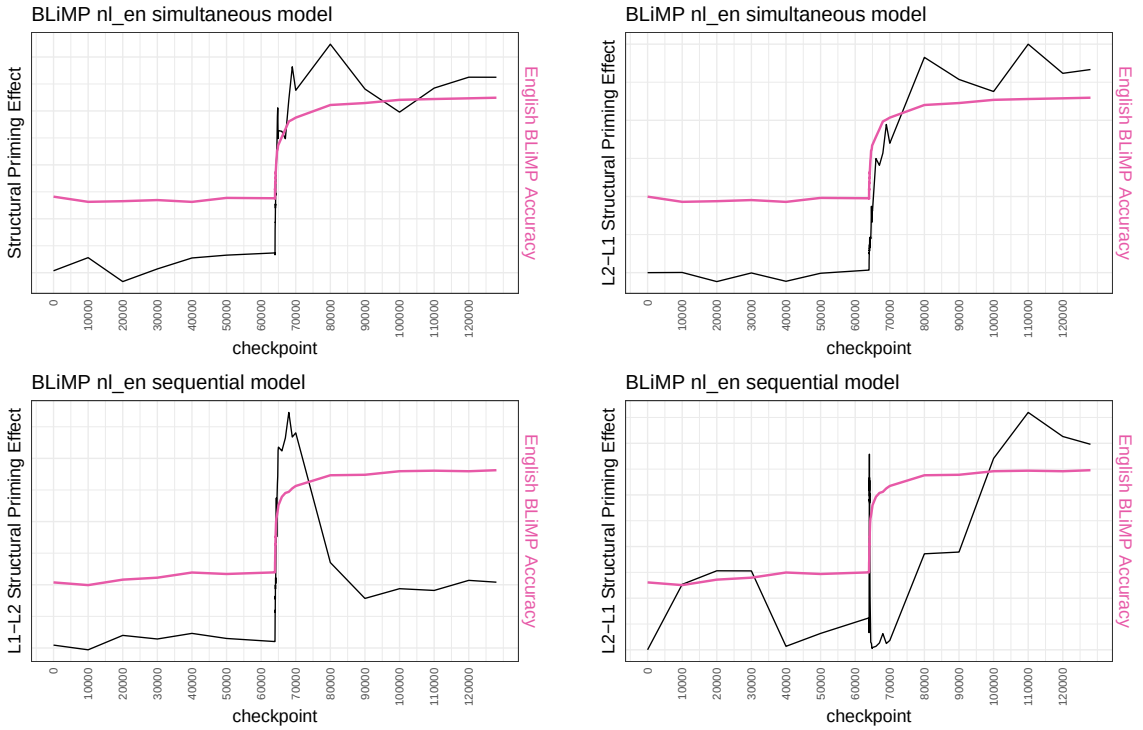

Figure D.18: Structural priming effect (black), plotted as the difference between match and mismatch conditions, and English BLiMP accuracy (pink) over the course of model training. Y-axes have been re-scaled for easier comparison.

D.6.2 Bernolet (2013)

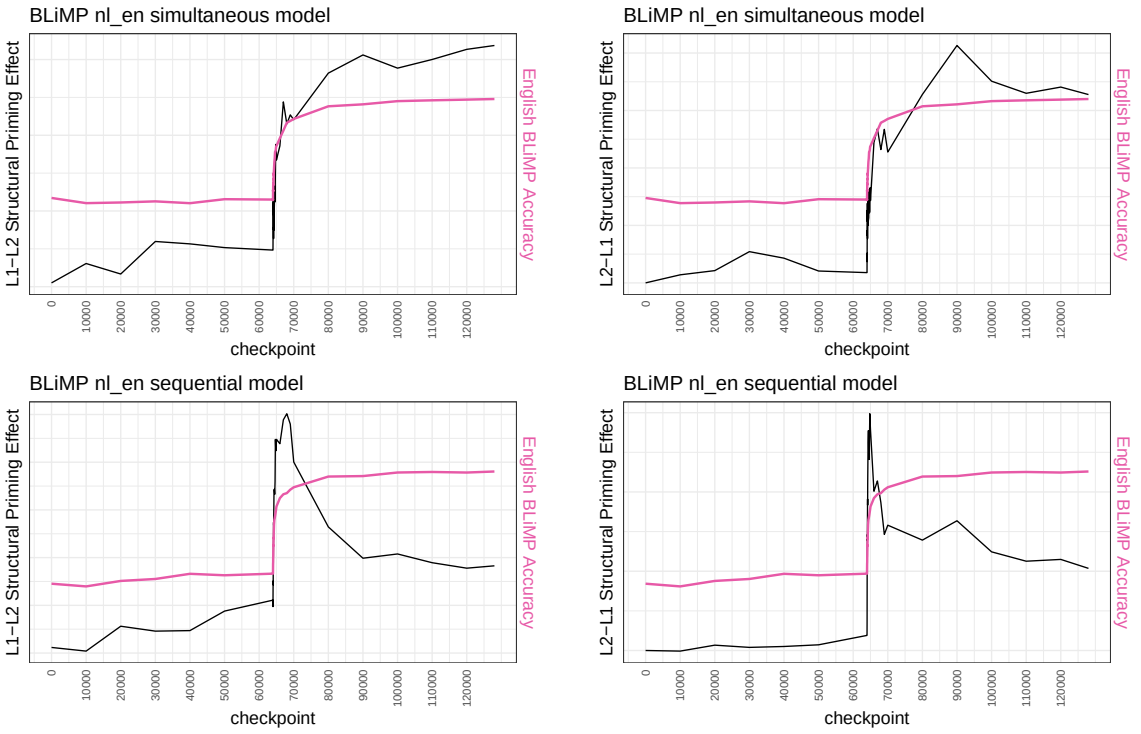

Figure D.19: Structural priming effect (black), plotted as the difference between match and mismatch conditions, and English BLiMP accuracy (pink) over the course of model training. Y-axes have been re-scaled for easier comparison.

D.6.3 Hartsuiker (2004)

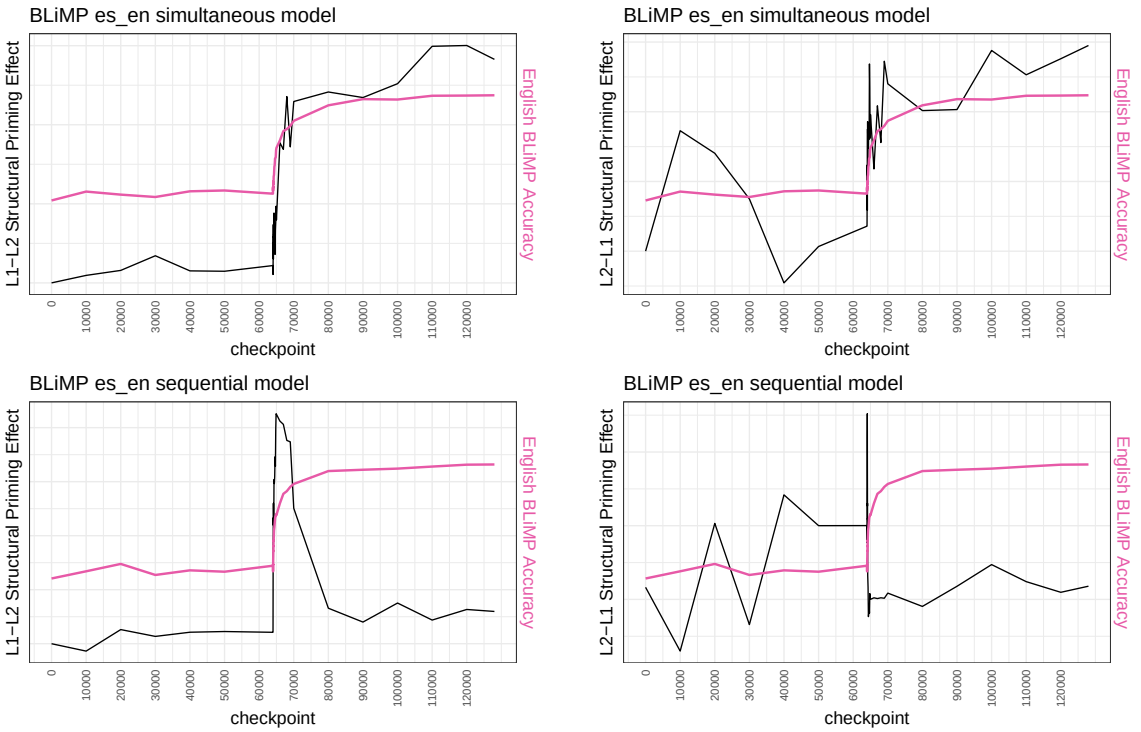

Figure D.20: Structural priming effect (black), plotted as the difference between match and mismatch conditions, and English BLiMP accuracy (pink) over the course of model training. Y-axes have been re-scaled for easier comparison.

D.6.4 Fleischer (2012)

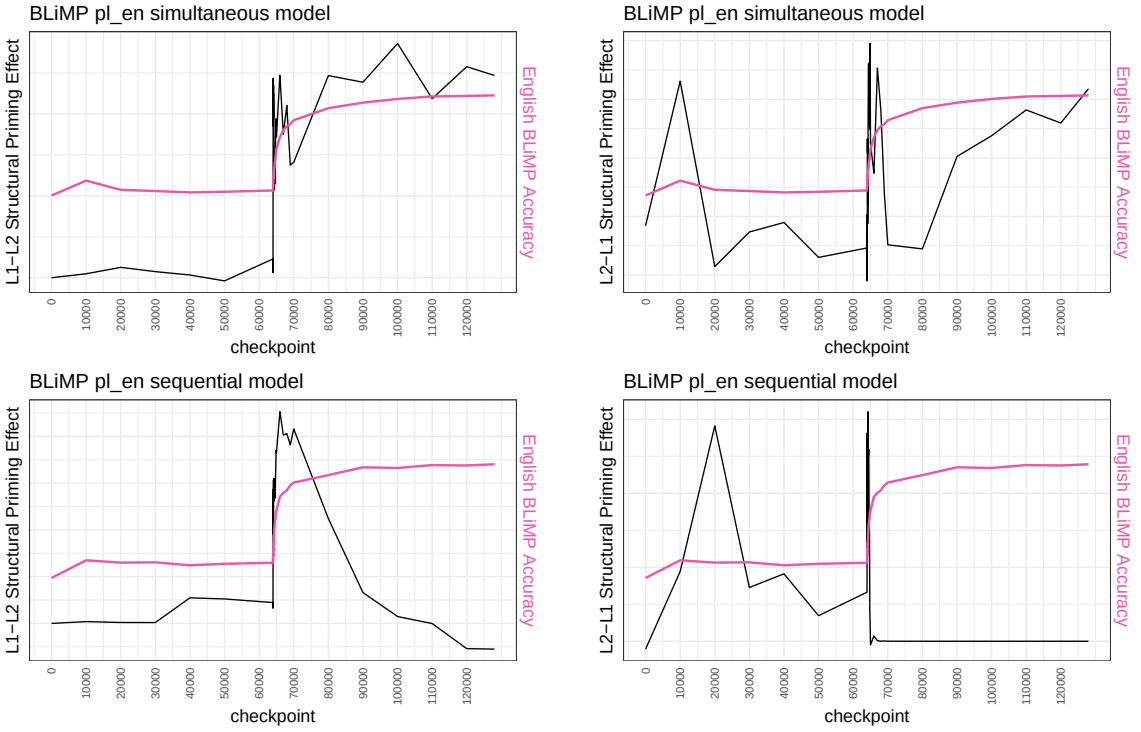

Figure D.21: Structural priming effect (black), plotted as the difference between match and mismatch conditions, and English BLiMP accuracy (pink) over the course of model training. Y-axes have been re-scaled for easier comparison.

D.6.5 Kotzochampou (2022)

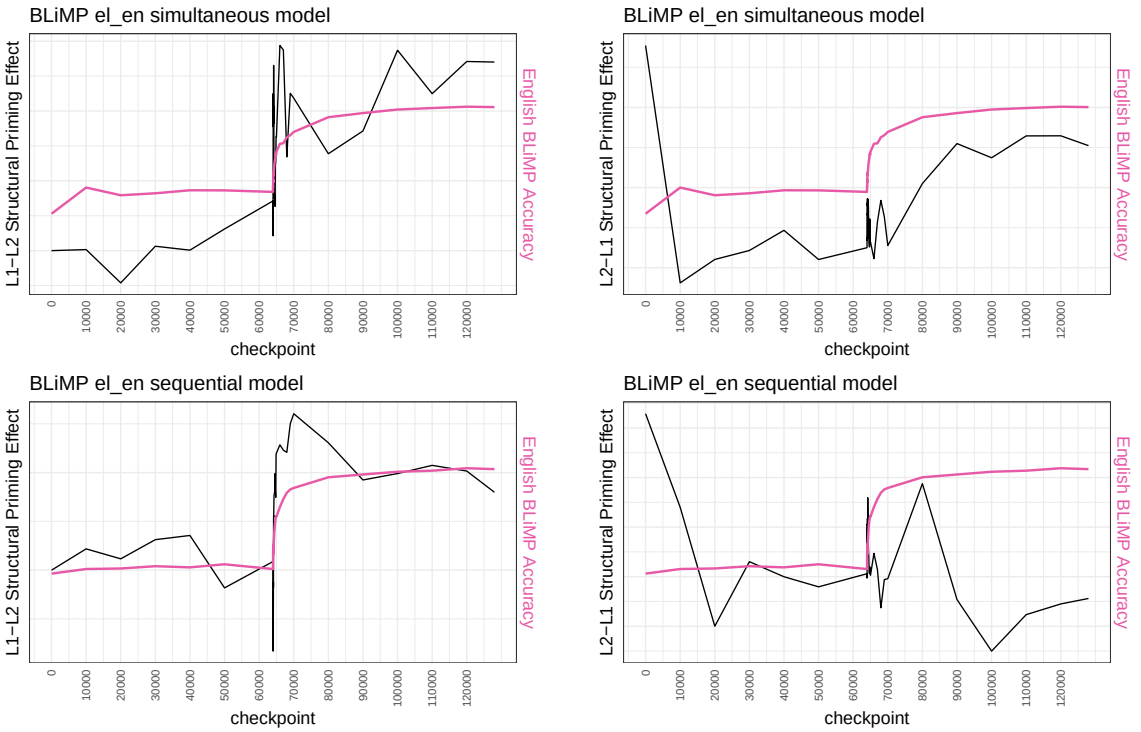

Figure D.22: Structural priming effect (black), plotted as the difference between match and mismatch conditions, and English BLiMP accuracy (pink) over the course of model training. Y-axes have been re-scaled for easier comparison.

D.7 Crosslingual LDA Classification Accuracy by Layer

D.7.1 Density Plots, Schoonbaert (2007)

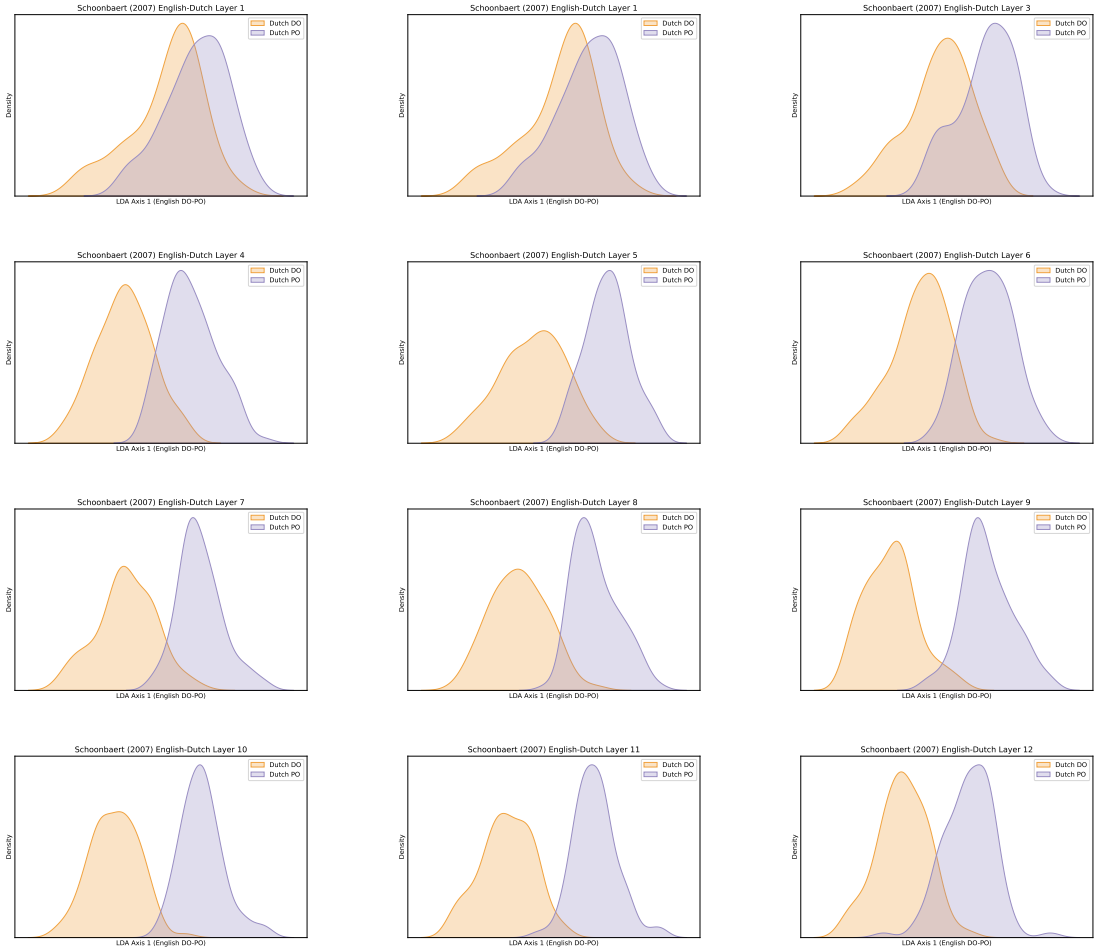

Figure D.23: Dutch DO (orange) and Dutch PO (purple) activations projected onto the LDA axis trained on English DO-PO activations. Each fact represents a different model layer.

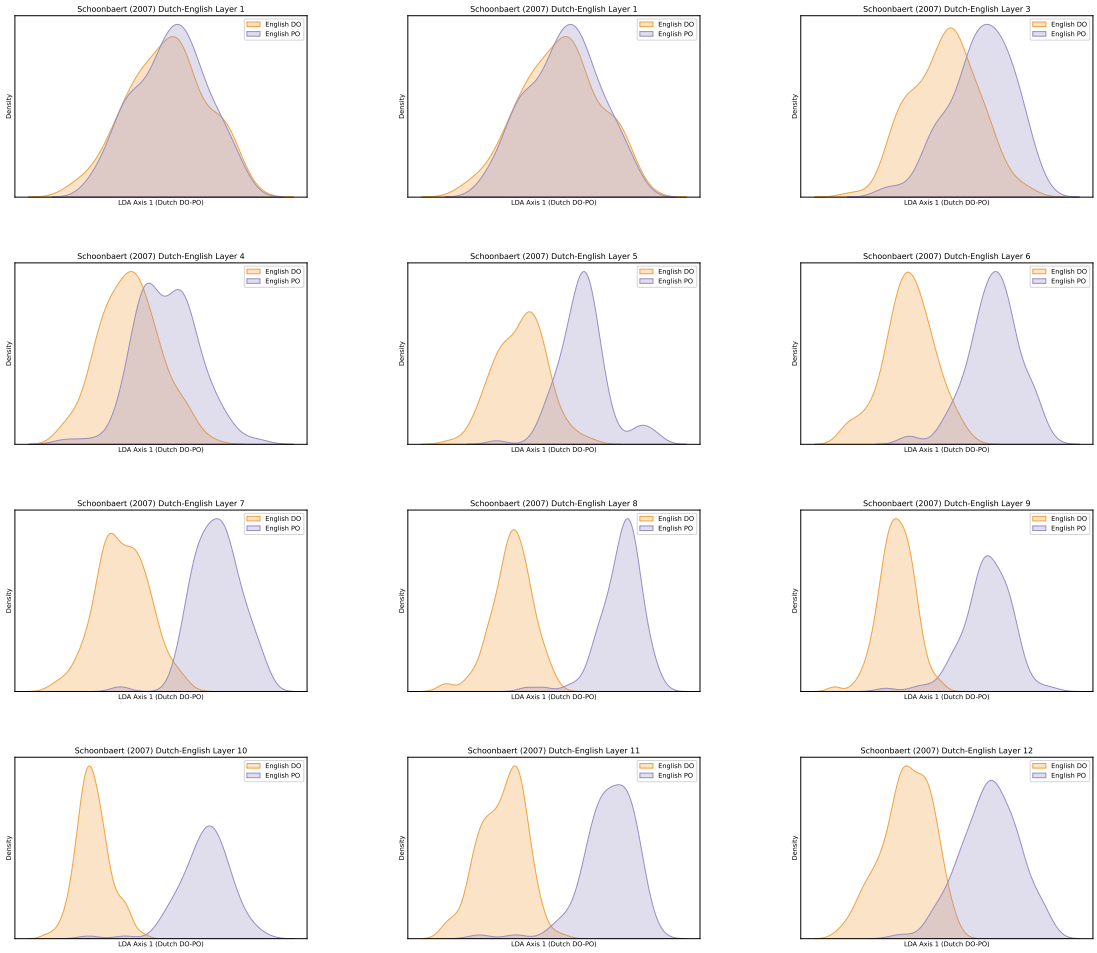

Figure D.24: English DO (orange) and English PO (purple) activations projected onto the LDA axis trained on Dutch DO-PO activations. Each fact represents a different model layer.

D.7.2 By Layer Accuracy

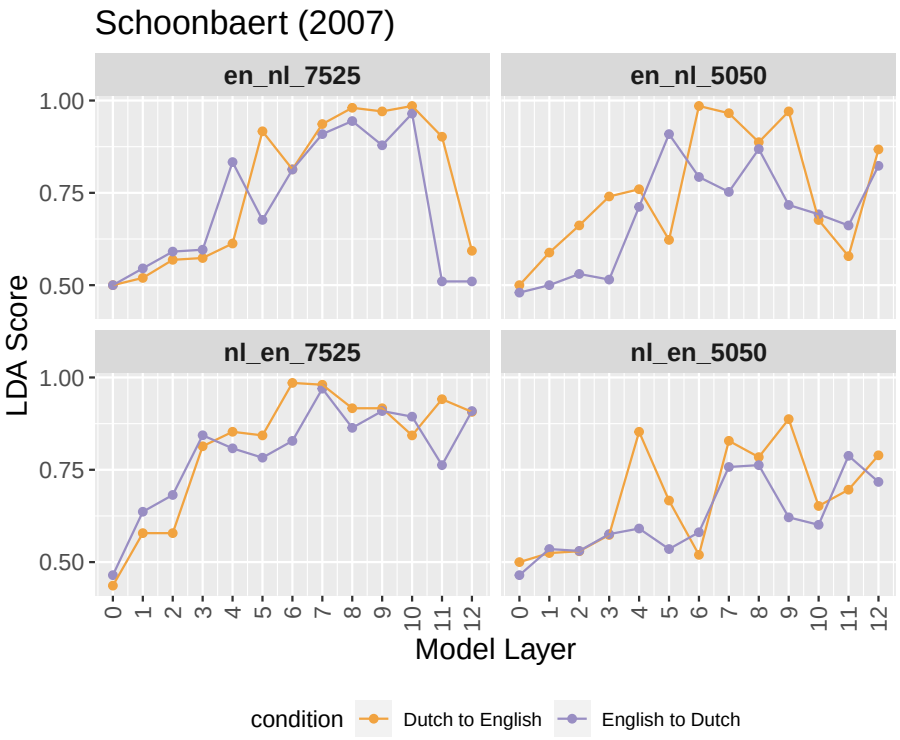

Figure D.25: Classification accuracy (y-axis) of the probe for each layer (x-axis). Orange represents the accuracy of the probe trained on Dutch at classification of English activations. Purple represents accuracy of probe trained on English for Dutch activations. Each facet represents a different model: en_nl_7525 is the English-Dutch simultaneous model. en_nl_5050 is the English-Dutch sequential model. nl_en_7525 is the Dutch-English simultaneous model. nl_en_5050 is the Dutch-English sequential model.

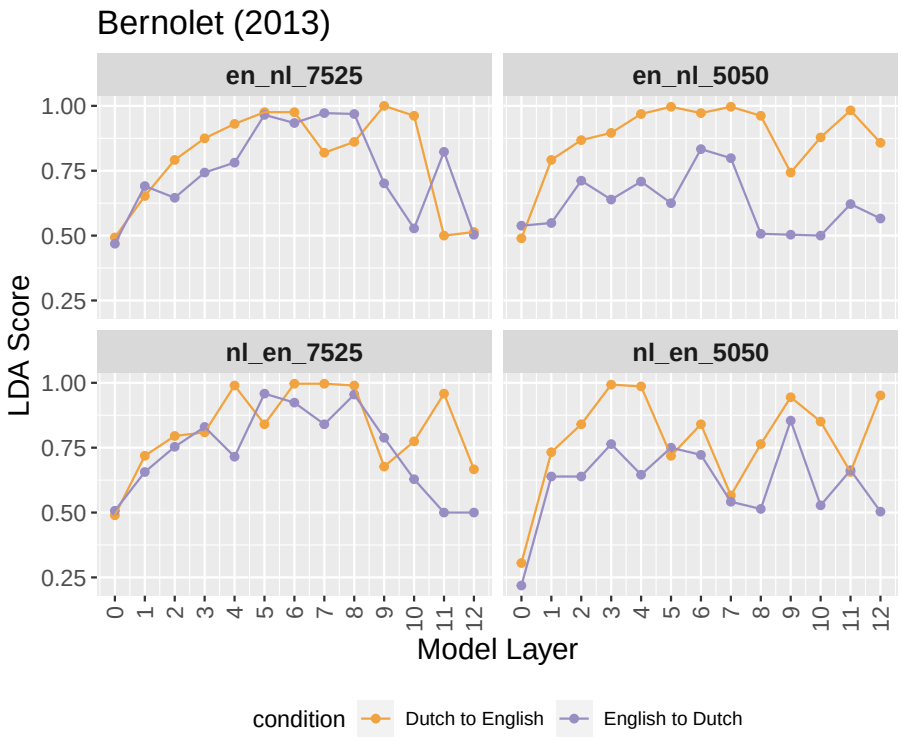

Figure D.26: Classification accuracy (y-axis) of the probe for each layer (x-axis). Orange represents the accuracy of the probe trained on Dutch at classification of English activations. Purple represents accuracy of probe trained on English for Dutch activations. Each facet represents a different model: en_nl_7525 is the English-Dutch simultaneous model. en_nl_5050 is the English-Dutch sequential model. nl_en_7525 is the Dutch-English simultaneous model. nl_en_5050 is the Dutch-English sequential model.

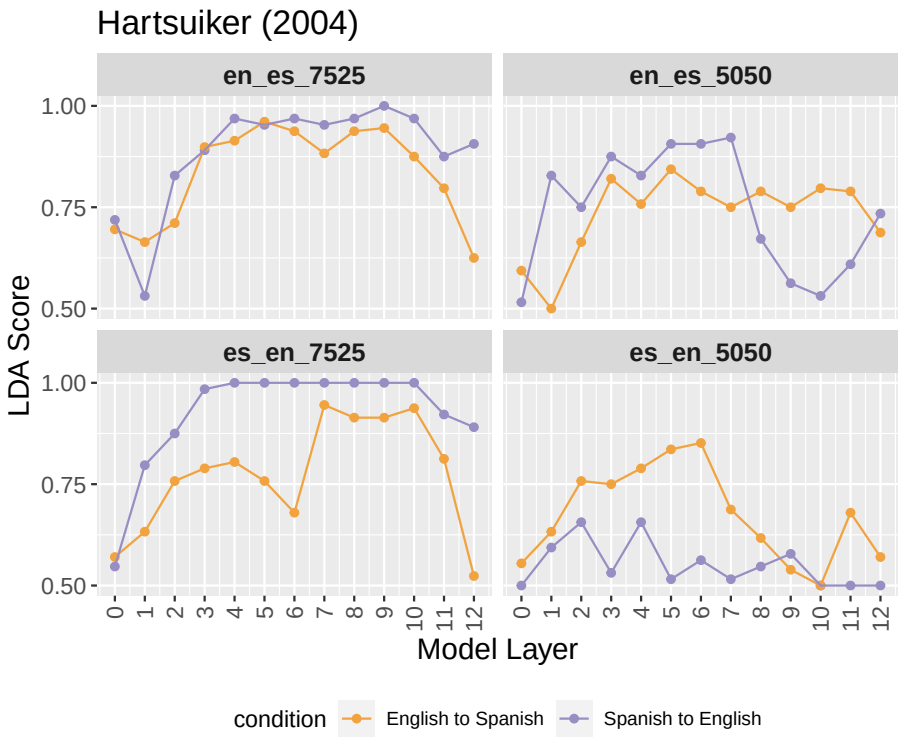

Figure D.27: Classification accuracy (y-axis) of the probe for each layer (x-axis). Orange represents the accuracy of the probe trained on Spanish at classification of English activations. Purple represents accuracy of probe trained on English for Spanish activations. Each facet represents a different model: en_es_7525 is the English-Spanish simultaneous model. en_es_5050 is the English-Spanish sequential model. es_en_7525 is the Spanish-English simultaneous model. es_en_5050 is the Spanish-English sequential model.

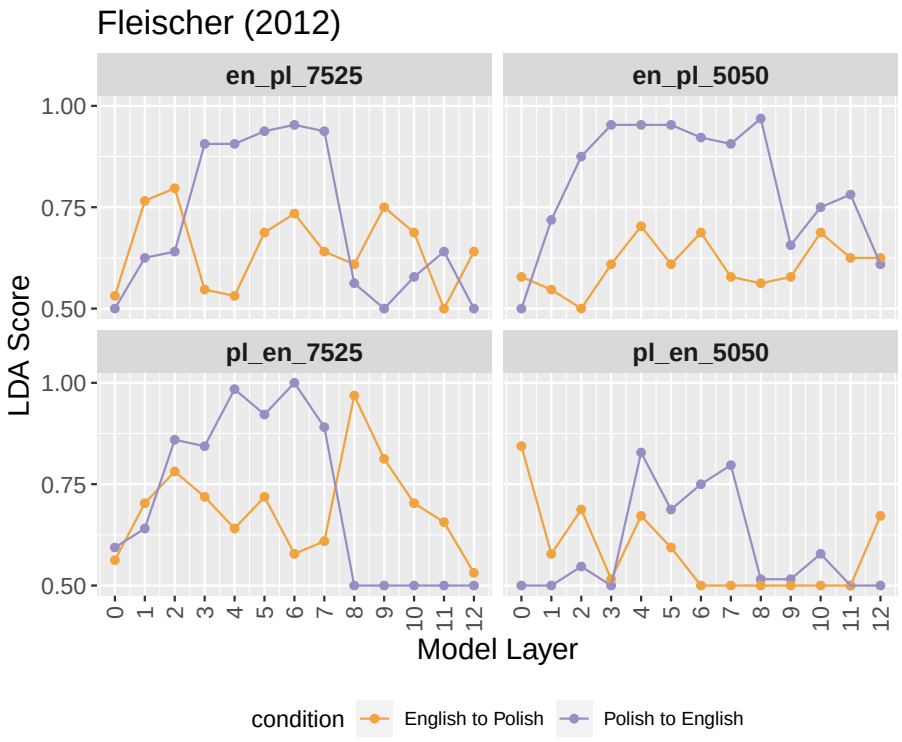

Figure D.28: Classification accuracy (y-axis) of the probe for each layer (x-axis). Orange represents the accuracy of the probe trained on Polish at classification of English activations. Purple represents accuracy of probe trained on English for Polish activations. Each facet represents a different model: en_pl_7525 is the English-Polish simultaneous model. en_pl_5050 is the English-Polish sequential model. pl_en_7525 is the Polish-English simultaneous model. pl_en_5050 is the Polish-English sequential model.

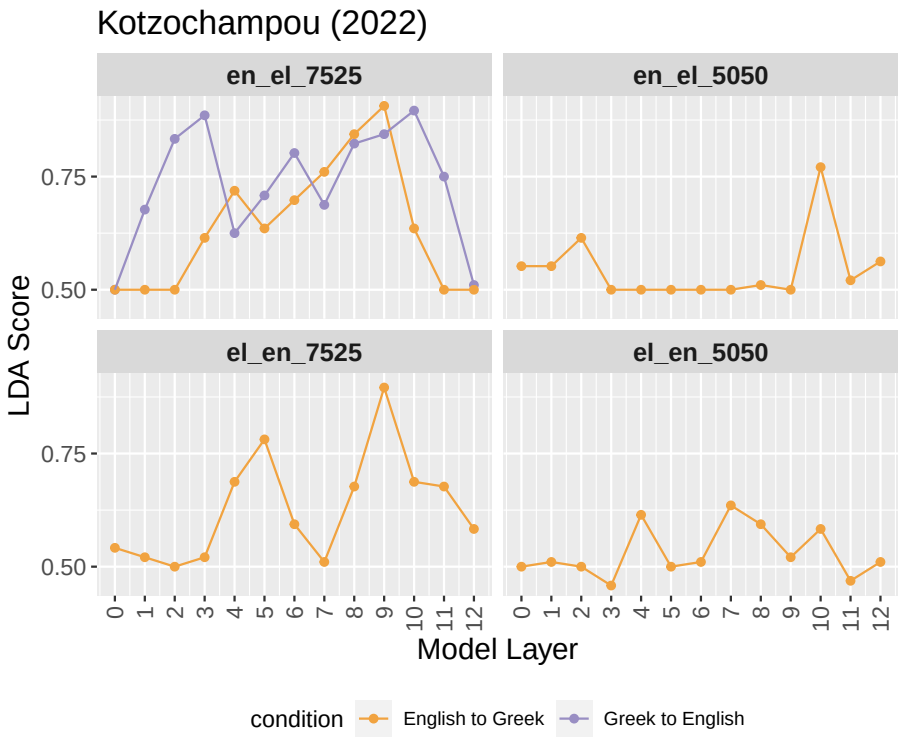

Figure D.29: Classification accuracy (y-axis) of the probe for each layer (x-axis). Orange represents the accuracy of the probe trained on Greek at classification of English activations. Purple represents accuracy of probe trained on English for Greek activations. Each facet represents a different model: en_es_7525 is the English-Spanish simultaneous model. en_es_5050 is the English-Spanish sequential model. es_en_7525 is the Spanish-English simultaneous model. es_en_5050 is the Spanish-English sequential model.

D.8 Cross-Constructional LDA Accuracy

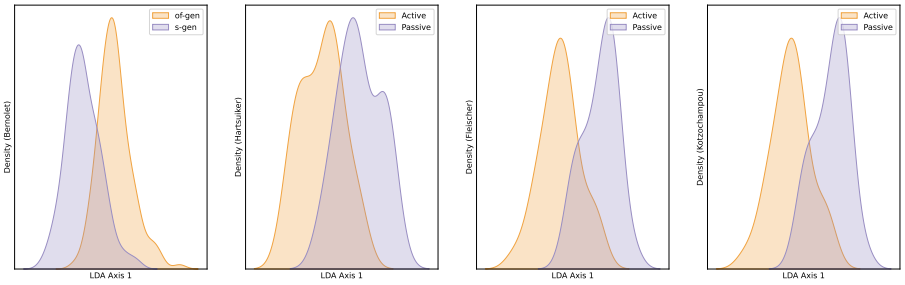

Figure D.30: English-Dutch simultaneous model; Layer 6

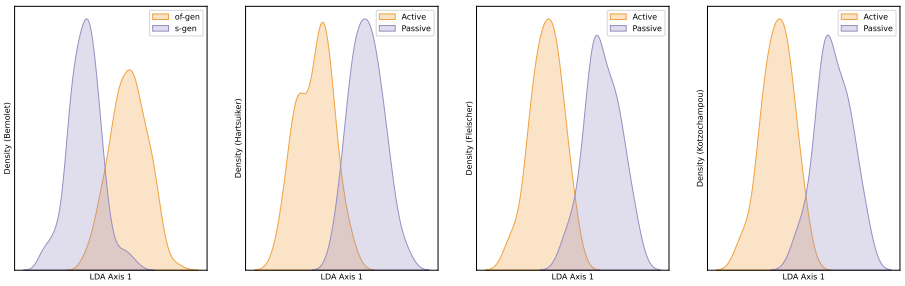

Figure D.31: English-Dutch simultaneous model; Layer 7

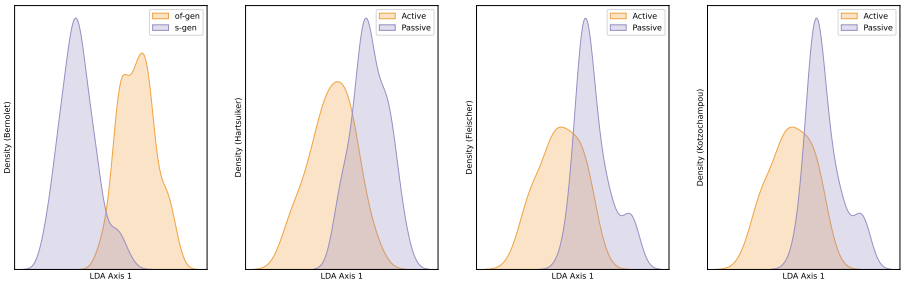

Figure D.32: English-Dutch simultaneous model; Layer 8

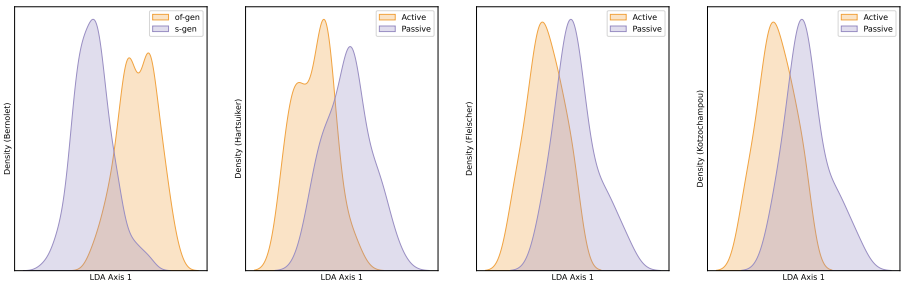

Figure D.33: English-Dutch simultaneous model; Layer 9

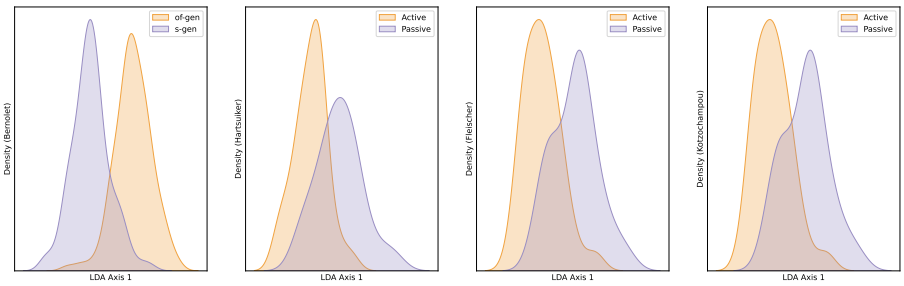

Figure D.34: English-Dutch simultaneous model; Layer 10

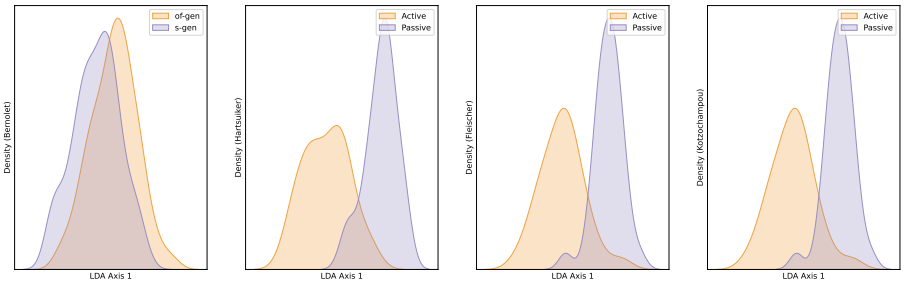

Figure D.35: English-Dutch simultaneous model; Layer 6

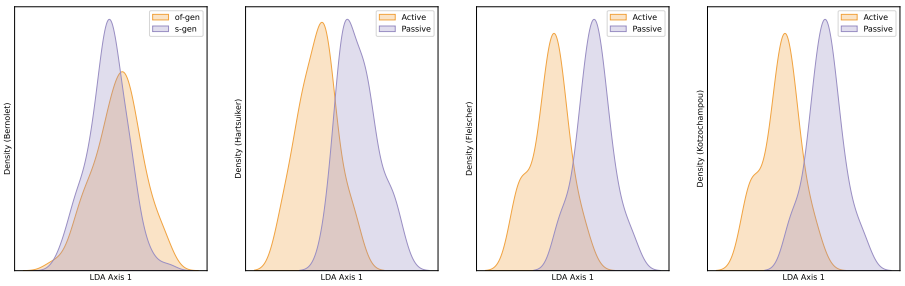

Figure D.36: Dutch-English simultaneous model; Layer 7

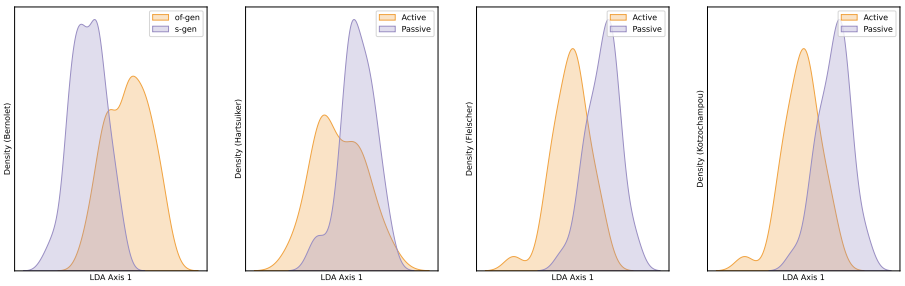

Figure D.37: Dutch-English simultaneous model; Layer 8

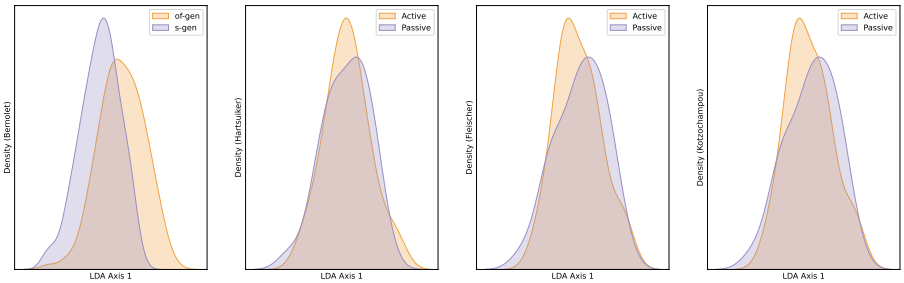

Figure D.38: Dutch-English simultaneous model; Layer 9

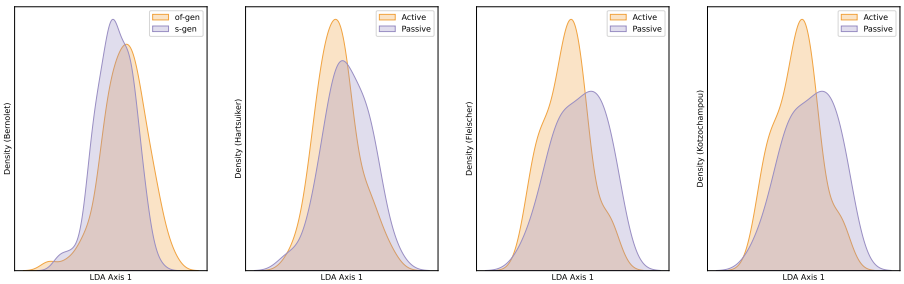

Figure D.39: Dutch-English simultaneous model; Layer 10

D.9 Modified Stimuli LDA Classification

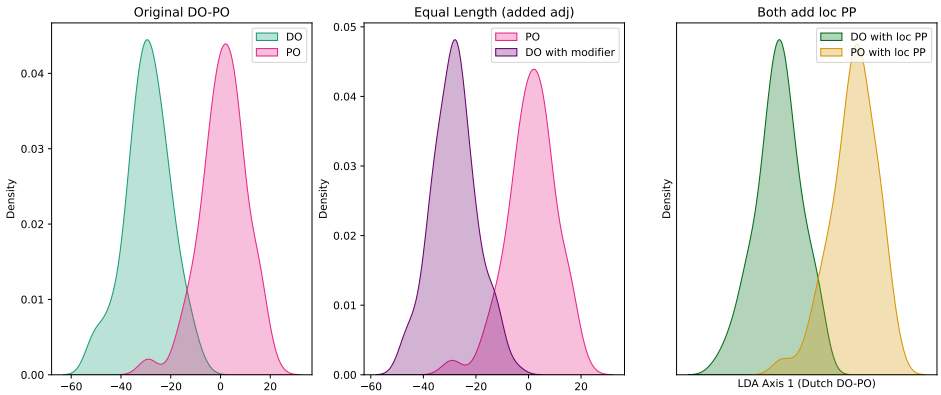

Figure D.40: LDA axis trained on Dutch DO-PO classification for English-Dutch simultaneous model; Layer 6

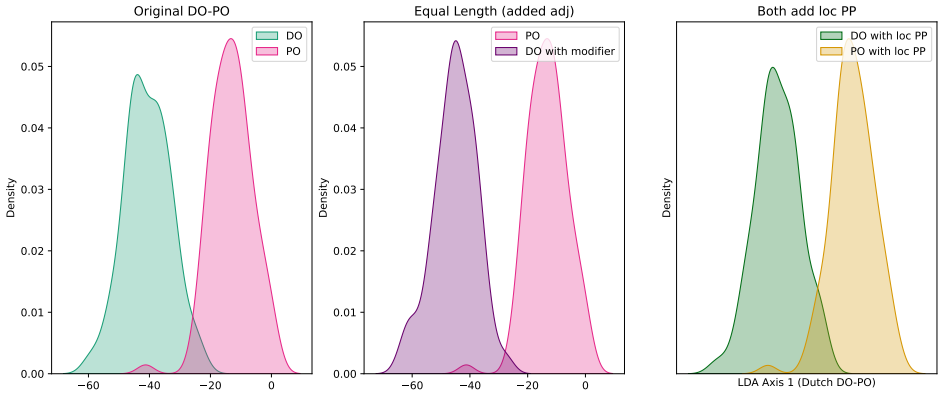

Figure D.41: LDA axis trained on Dutch DO-PO classification for English-Dutch simultaneous model; Layer 7

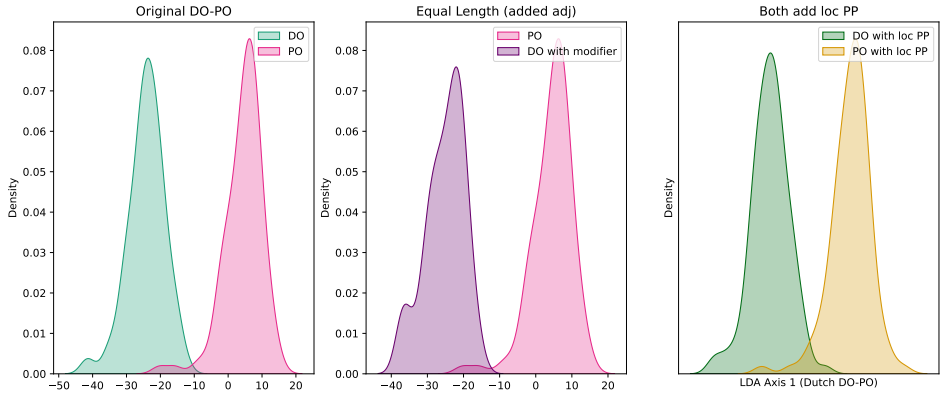

Figure D.42: LDA axis trained on Dutch DO-PO classification for English-Dutch simultaneous model; Layer 8

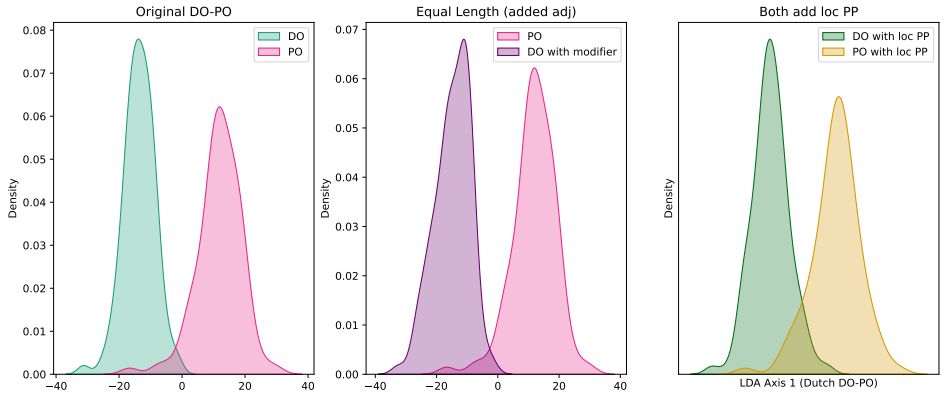

Figure D.43: LDA axis trained on Dutch DO-PO classification for English-Dutch simultaneous model; Layer 9

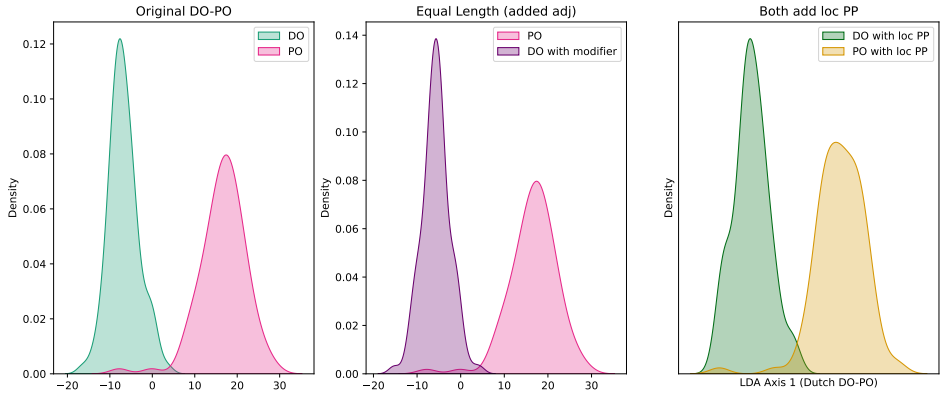

Figure D.44: LDA axis trained on Dutch DO-PO classification for English-Dutch simultaneous model; Layer 10

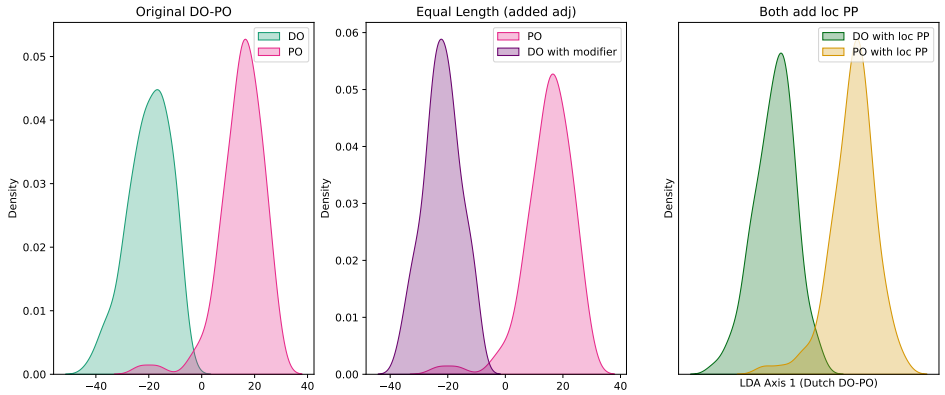

Figure D.45: LDA axis trained on Dutch DO-PO classification for English-Dutch simultaneous model; Layer 6

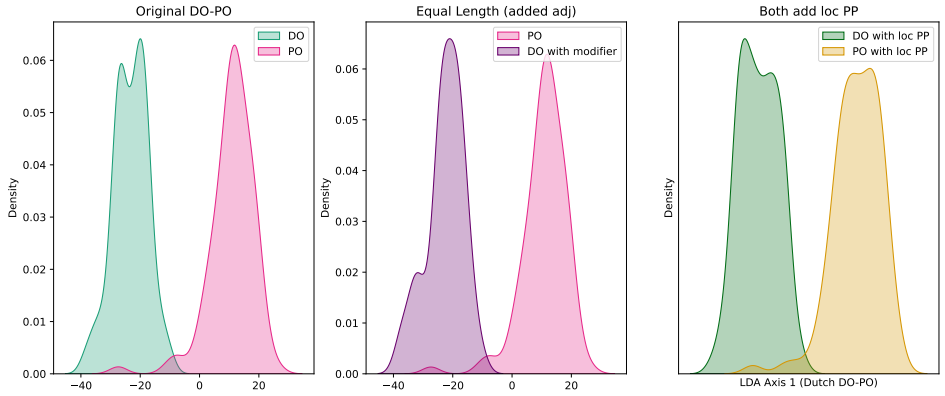

Figure D.46: LDA axis trained on Dutch DO-PO classification for English-Dutch simultaneous model; Layer 7

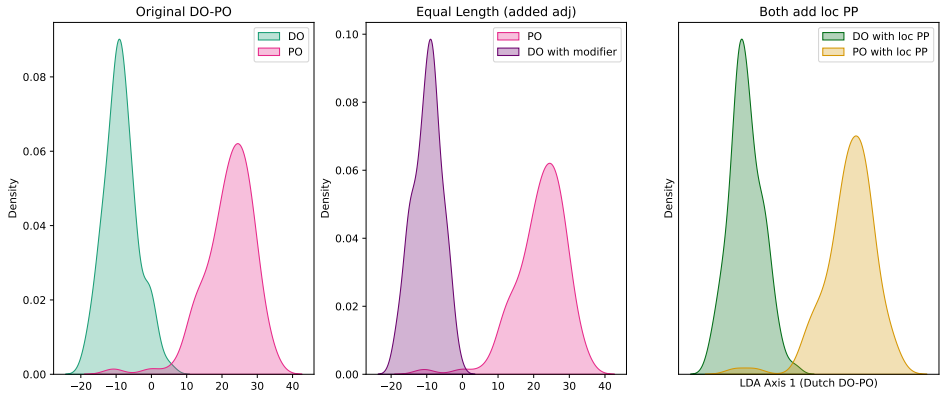

Figure D.47: LDA axis trained on Dutch DO-PO classification for English-Dutch simultaneous model; Layer 8

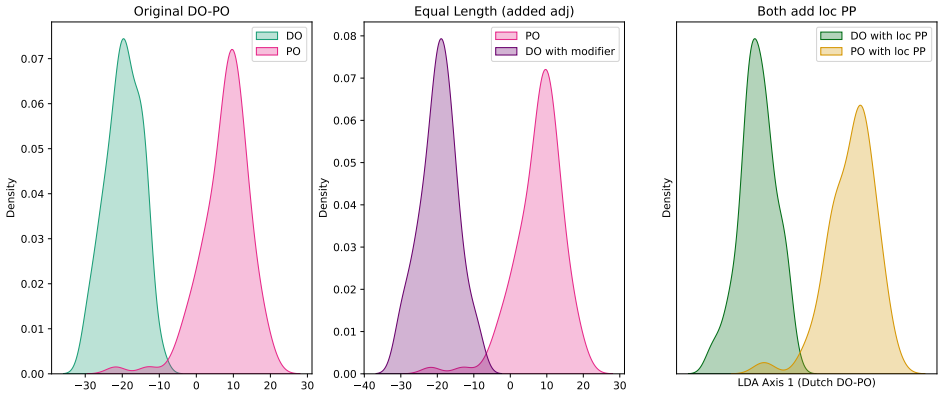

Figure D.48: LDA axis trained on Dutch DO-PO classification for English-Dutch simultaneous model; Layer 9

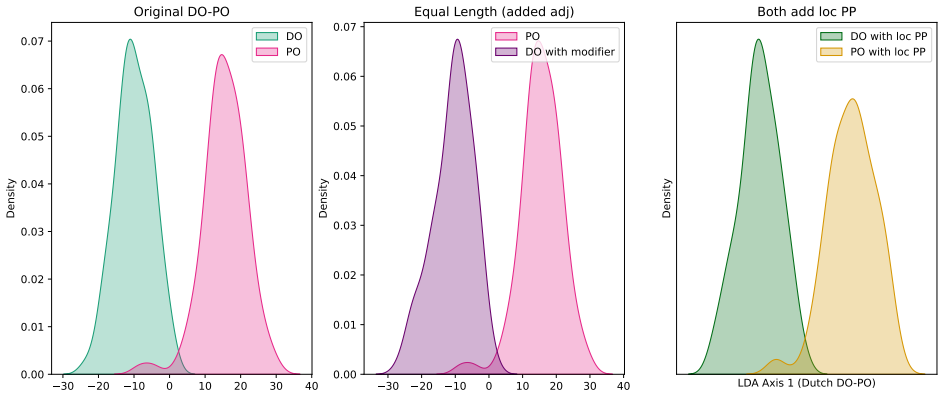

Figure D.49: LDA axis trained on Dutch DO-PO classification for English-Dutch simultaneous model; Layer 10