MODELING SYNTACTIC AMBIGUITY WITH DEPENDENCY PARSING

by

BERTA FRANZLUEBBERS

(Under the Direction of John T. Hale)

ABSTRACT

A generative incremental dependency parser enhanced with sequence encodings from a large language model was used to calculate a syntactic surprisal measure in order to analyze the ambiguity present when listening to an audiobook. This metric was correlated with BOLD fMRI signal, confirming the hypothesis that derivations with low predicted probability require greater effort to understand. This surprisal metric was validated at various levels of parallel processing, providing evidence that increasing the level of parallelism creates a significantly better predictor for the data, up to a threshold which includes most viable derivations. The brain regions associated with increased activation for higher levels of parallelism across English and Chinese were identified as bilateral superior temporal gyrus activations.

INDEX WORDS: [Neurolinguistics, Cognitive Neuroscience, Linguistic Syntax, fMRI, GLM]

MODELING SYNTACTIC AMBIGUITY WITH DEPENDENCY PARSING

by

BERTA FRANZLUEBBERS

A.B., University of Georgia, 2016
B.S., University of Georgia, 2016

A Thesis Submitted to the Graduate Faculty of the
University of Georgia in Partial Fulfillment of the Requirements for the Degree.

MASTER OF SCIENCE

ATHENS, GEORGIA

2023

MODELING SYNTACTIC AMBIGUITY WITH DEPENDENCY PARSING

by

BERTA FRANZLUEBBERS

Major Professor:	John Hale
Committee:	Michael Covington Frederick Maier

Electronic Version Approved:

Ron Walcott
Dean of the Graduate School
The University of Georgia
August 2023

ACKNOWLEDGMENTS

I would like to thank everyone involved in data collection, preprocessing, and annotation, without which this project would not be possible.

I would also like to thank Donald Dunagan for his much appreciated help with fMRI analysis code, and Dr. Jan Buys for making his code available, and his helpful suggestions.

I would like to thank all the members of my committee: Dr. Michael Covington for his valuable input and inspiration to consider SUD, Dr. Fredrick Maier for his comments and encouragement to enroll in the MSAI program, and Dr. John Hale for his advisement and encouragement throughout this project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

§ 1.1 Overview

This work aims to analyze human sentence comprehension in the brain by modeling syntactic ambiguity using dependency parsing. While listening to speech, humans process the words they hear into a meaningful representation. Building this semantic representation involves combining individual word meanings into phrases, according to the syntactic relationships between words. While there are different theories of syntax, one of the most fundamental analyses is how individual words relate to one another. This is captured in Dependency Grammar by the head - dependent relationship between words (Tesnière, 1959). However, determining the correct relationships is not always straightforward. At times a sentence can be globally ambiguous, resulting in two or more different interpretations. More commonly, multiple representations are possible because the entire sentence has not yet been heard. Human sentence comprehension is incremental, and predictions about the rest of the sentence are continually being made and updated over time (DeLong et al., 2014; Kamide et al., 2003; Marslen-Wilson, 1973; Tanenhaus et al., 1995). A discrepancy between this prediction and the updated analysis after new information is observed corresponds to high comprehension difficulty. In addition, some alternate interpretations are simply not plausible, and would be predicted with low probability. If a sentence parsing strategy claims to be cognitively plausible, then its predictions about the relative difficulty of disambiguating between interpretations must correlate with human behavior. In this study, fMRI data collected while participants listen to an audiobook serves as the ground truth against which the complexity predictions of a dependency parser are evaluated.

§ 1.2 Motivation

This approach follows the language comprehension studies reviewed in Hale et al., 2022 in evaluating a word-by-word complexity metric against brain data. However, most studies are limited by their analysis of metrics related to the top ranked syntactic analysis. Notable exceptions are Hale et al., 2018 and Crabbé et al., 2019 which consider the dynamic parsing process and include characteristics of the top derivations included in the beam in their processing predictions. These studies both use Recurrent Neural Network Grammars (Dyer et al., 2016) to indicate the phrase structure. In contrast, this work relies on Dependency Grammar, which Boston et al., 2011 showed is able to predict sentence comprehension difficulty via eye fixation data. In addition, Boston et al., 2011 also varied the number of alternate derivations taken into account, and found evidence in favor of ranked parallel processing. Although ambiguity in natural language is a given, authorities differ on the parsing strategy of the human system. The debate between ranked parallel and serial processing of only the best derivation, with potential for reanalysis, has been discussed since Fodor et al., 1974. Although some such as Van Gompel et al., 2005 are in favor of serial processing, others such as Jurafsky, 1996 claim that humans employ ranked parallel processing. The extent to which ambiguity processing, which requires analysis of alternative derivations, occurs is still in question across languages.

In addition to clarifying the amount of ambiguity resolution required for language understanding using Dependency Parsing, another motivation for this work is to incorporate the parsing improvements possible with large language models. Modern AI models are trained on larger datasets, more accurately including distributional linguistic information; however, they pose a difficulty with regards to explainability. Instead of deriving a metric directly from the output of a large language model, or correlating next word predictions with fMRI data, as in Schrimpf et al., 2021 and Caucheteux and King, 2022, this work uses uses large language model (LLM) embeddings to train an incremental parser, providing a probability for each possible derivation. This provides a level of explanation valuable when analyzing the linguistic structures involved in the processing of ambiguity.

§ 1.3 Contributions

This ability to calculate a probability for each derivation from a generative dependency parser was developed by Buys and Blunsom, 2018, who used an RNN sequence model to encode the sentences. This work replaces the RNN with pretrained LLM embeddings, which contribute to a significant increase in accuracy, while still relying on the exact marginalization framework provided by Buys and Blunsom, 2018.

In addition, the incremental, left-to-right nature of the parser was exploited in order to calculate the probabilities for sentence-medial, partially complete derivations. Although including these derivations increases the total number under consideration factorially, the top k were calculated efficiently by using a new summation layer and the original dynamic program.

With these derivations in hand, various applications of surprisal as a complexity metric were examined, resulting in a syntactic surprisal metric that takes advantage of increased accuracy from a generative model, without explicitly including word generation probabilities in order to focus on syntactic differences.

This surprisal metric was validated at various levels of parallel processing, using the *Little Prince* fMRI dataset (Li et al., 2022). Since increasing k resulted in a significantly better predictor for the data, this analysis provides new evidence in favor of parallel processing, at low values of k.

In comparing surprisal at various levels of parallel processing, brain regions associated with increased activation for higher levels of parallel processing across English and Chinese were identified, namely the superior temporal gyrus in both left and right hemispheres. This significant difference in activation implies that these regions are involved in disambiguating among competing syntactic derivations. The bilateral activation coincides with the expectation to see greater right hemisphere activation in naturalistic listening data, compared with studies using individual sentences, as well as with the conclusions of Mason et al., 2003 who predict bilateral activation among homologous regions, when processing difficulty exceeds a threshold.

§ 1.4 Thesis Structure

Before detailing these contributions, we begin by reviewing the related literature in Chapter 2, which includes background on Dependency Parsing and its annotation schemas, as well as psycholinguistic studies related to syntactic processing.

Chapter 3 discusses the development of the dependency parser, the choice of complexity metric to reflect the ambiguity level at each word, and finally turns to the evaluation of this metric against fMRI data.

Then, Chapter 4 presents and discusses the results, while Chapter 5 concludes.

CHAPTER 2

BACKGROUND AND RELATED WORK

§ 2.1 Dependency Parsing

The theory of Dependency Grammar posits that syntactic structure can be summarized by a set of relations between words in the sentence. These relations are binary and asymmetric (Tesnière, 1959). For a corpus of sentences in any language, each sentence has a dependency tree containing labeled, directed arcs between each word and its dependents.

An example dependency tree is given in Fig. 2.1. The arcs point from *head* to *dependent*, where each word has no more than one head. The ROOT symbol is included so that a complete dependency graph (which can then be called tree) has exactly one head per word in the sentence. The ROOT symbol is not allowed to have a head. A further condition is that of acyclicity: there is no subset of dependency arcs such that the head of a word is also its dependent (even with intervening words). That is, a graph G is acyclic when there is no (non-empty) subset of arcs (dependent, label, head) satisfying: $\{(i_0, l_1, i_1), (i_1, l_2, i_2), ..., (i_{k-1}, l_k, i_k)\}$ such that $i_0 = i_k$ (Nivre, 2008).

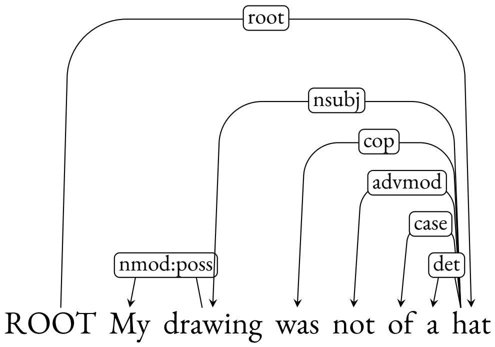

Figure 2.1: A dependency tree for a sentence from *The Little Prince*, according to the Universal Dependencies (UD) framework

However, there are different ideas about the criteria for deciding between which two words to draw a dependency relation, and which word in the pair should be the head. De Marneffe and Nivre, 2019 review a summary of common characteristics between a head H and a dependent D in a construction C:

Table 2.1: Characteristics for identifying a syntactic relationship between a head H and a dependent D in a construction C (De Marneffe and Nivre, 2019; Hudson, 1990; Zwicky, 1985)

1	H determines the syntactic category of C and can often replace C.
2	H determines the semantic category of C; D provides semantic specification.
3	H is obligatory; D may be optional.
4	H selects D and determines whether D is obligatory or optional.
5	The form of D depends on H (agreement or government).
6	The linear position of D is specified with reference to H.

Some dependency relation are uncontroversial: verbs are always considered the head of their arguments, even though the head cannot replace the construction (1), but the verbal head selects its arguments (4). Other constructions follow characteristic (1) and are also uncontroversial, e.g. adjectives are dependents on the noun they modify.

However, other constructions have conflicting explanations. According to Mel'cuk et al., 1988, different relations can be drawn based on semantic, syntactic, or morphological criteria. Even syntactic dependencies could be different based on deep or surface representations.

One particular difficulty is the representation of function words (e.g. auxiliaries, prepositions, determiners). For these words, the properties for identifying a syntactic head (Table 2.1) are often distributed across multiple words. For example, subject-verb agreement (5) is marked on auxiliary verbs in English, but valency (4) is determined by the main verb.

The Universal Dependencies Corpora makes the decision to categorize content words as heads instead of function words (De Marneffe et al., 2021). This is justified by setting the goal of making the dependency relations most comparable across languages. In many languages function words are not used (to the same extent), or morphological inflection is used instead of separate function words. With the UD framework, the core dependencies remain the same when the same sentence is translated into different languages, when the content words are used as heads of any function words present.

In contrast, the Surface-Syntactic Universal Dependencies (SUD) annotation schema uses function words as heads, which corresponds to characteristic (1) of heads, determining the syntactic distribution of the construction (Gerdes et al., 2021).

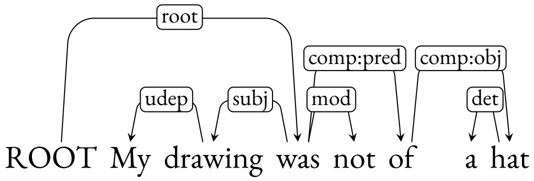

Figure 2.2: A dependency tree for a sentence from *The Little Prince*, according to the Surface-Syntactic Universal Dependencies (SUD) framework

§ 2.3 Transition System

When analyzing algorithms that can generate a dependency tree, there are two classes which differ in the storage system for partially processed tokens (tokens that have been read and are being considered for attachment). The first type uses open lists, which allows all (including non-projective) dependency structures to be processed (Covington, 2001; Nivre et al., 2007). In contrast stack-based algorithms are typically restricted to projective graphs (Kudo and Matsumoto, 2002; Yamada and Matsumoto, 2003; Nivre, 2003). Projective dependency graphs are defined by Mel'cuk et al., 1988 such that arcs between words do not cross and no arc covers the top node.

For stack-based algorithms, a parser configuration (c) is defined as a stack, buffer, and dependency arcs already drawn $c = (\sigma, \beta, A)$. The transition system is a set of maps from configuration to configuration. The transition system processes the words in each sentence $w_{0:n}$ from left to right, where w_0 is the root symbol, ROOT, and w_n is the end-of-sentence symbol, EOS. To satisfy the condition of projectivity, the ROOT symbol is the top node, which is not permitted to have a head.

The set of possible transitions varies depending on the specific parser, but all transitions fall within the shift-reduce parsing system, where *Shift* transitions move the next word into focus (i.e. onto the stack), and *Reduce* transitions remove a word from the stack, normally as the result of drawing an arc between two words. The order by which the arcs between words are drawn can be classified using the terms top-down and bottom-up, but these are defined differently from their standard meaning in context-free parsing. Top-down construction of dependency graphs does not involve any prediction of nodes, since the dependency graph does not have non-terminal nodes, and the terminal nodes are given by the input string. In terms of context-free parsing, all dependency parsing algorithms would therefore be bottom-up. However, for dependency parsing, the term top-down is defined as attaching a dependent to its head before the dependent is attached to all of its dependents. Conversely, bottom-up means that a dependent must be attached to its head before that head is attached to its head (Nivre, 2004).

The arc-eager system (Nivre, 2003) draws arcs as soon as possible, even if the dependent has not yet been attached to its own dependents. This system has 4 possible transitions, where *Shift* and *Right-Arc* both shift the word from the buffer to the stack, but *Right-Arc* also draws a dependency arc from the word on top of the stack to the next word on the buffer, before shifting the word. The other possible transitions, *Left-Arc* and *Reduce* are contingent on the status of the word on top of the stack. If the word is already headed, it can be popped from stack with (*Reduce*), while if it has no head, an arc can be drawn from the next word on the buffer to the word on top of the stack (*Left-Arc*). Drawing arcs as soon as possible reduces the number of words on the stack, making this system the most incremental. However, this also means that left-dependents are processed bottom-up, and right-dependents are processed top-down.

In contrast, a true bottom-up system requires all dependents of a word to be found before the word can find its head. This system can be arc-standard (Nivre, 2004) or arc-hybrid (Kuhlmann et al., 2011). The arc-standard system has a simple set of allowed transitions: *Shift* moves the next word from the buffer to the stack, while *Left-Arc* and *Right-Arc* are symmetric, that is both draw arcs between the first and second elements of the stack, and only of the direction of the arc distinguishes the transitions (e.g. *Left-Arc* draws an arc from the second stack element to the first stack element). Finally, the arc-hybrid system is named for its combination of transitions from the arc-standard and arc-eager systems. The *Shift* and *Right-Arc* transitions are defined identically to the arc-standard system. The *Left-Arc* transition is the same as the arc-eager system, however this transition can always apply. This is because all words on the stack do not have a head, which simplifies the record keeping when using this system. As noted, both arc-standard and arc-hybrid systems are bottom-up, although the *Left-Arc* transition is defined differently. However, this does not change the order in which arcs are drawn. The difference simply comes down to applying a *Shift* action before or after the left-arc is drawn.

A sample transition action sequence for the sentence in Fig. 2.1 is shown in Table 2.2, which uses the UD framework. In contrast, Table 2.3 shows the same sentence from Fig. 2.2, which uses the SUD framework. It is apparent in this example, and generally true for English, that SUD has a shorter average distance between words linked by a dependency.

Table 2.2: Arc-hybrid transition system derivation for the UD sentence from Fig. 2.1

Stack σ	Index β	Prediction
ROOT	My	Shift(My)
ROOT, My	drawing	Left-Arc(nmod:poss)
ROOT	drawing	Shift(drawing)
ROOT, drawing	was	Shift(was)
ROOT, drawing, was	not	Shift(not)
ROOT, drawing, was, not	of	Shift(of)
ROOT, drawing, was, not, of	a	Shift(a)
ROOT, drawing, was, not, of, a	hat	Left-Arc(det)
ROOT, drawing, was, not, of	hat	Left-Arc(case)
ROOT, drawing, was, not	hat	Left-Arc(advmod)
ROOT, drawing, was	hat	Left-Arc(cop)
ROOT, drawing	hat	Left-Arc(nsubj)
ROOT	hat	Shift(hat)
ROOT, hat	EOS	Right-Arc(root)
ROOT	EOS	reduce (p=1)

The comparison between UD and SUD in Tables 2.2 and 2.3 illustrates the result of shorter average distance between words linked by relations: the stack is shorter on average for the SUD framework. This observation that working memory requirements are greater for longer dependency distances is the reason for a cross-linguistic preference for shorter distance between dependencies. Liu, 2008 analyzed corpora from 20 languages and found that the average dependency distance only varies slightly, and is always smaller than four, which is within working memory constraints (Cowan, 2001). Futrell et al., 2015 also found that dependency length is smaller than a random baseline for 37 languages. Nevertheless, long-distance dependencies are still possible, and even common in some languages, since there are competing preferences which may result in requiring longer average dependency distance (Hahn & Xu, 2022).

Table 2.3: Arc-hybrid transition system derivation for the SUD sentence from Fig. 2.2

Stack σ	Index β	Prediction
ROOT	My	Shift(My)
ROOT, My	drawing	Left-Arc(udep)
ROOT	drawing	Shift(drawing)
ROOT, drawing	was	Left-Arc(subj)
ROOT	was	Shift(was)
ROOT, was	not	Shift(not)
ROOT, was, not	of	Right-Arc(mod)
ROOT, was	of	Shift(of)
ROOT, was, of	a	Shift(a)
ROOT, was, of, a	hat	Left-Arc(det)
ROOT, was, of	hat	Shift(hat)
ROOT, was, of, hat	EOS	Right-Arc(comp:obj)
ROOT, was, of	EOS	Right-Arc(comp:pred)
ROOT, was	EOS	Right-Arc(root)
ROOT	EOS	reduce (p=1)

§ 2.4 Deduction System

Although the number of transitions required to parse a sentence is exponential on the number of words in the sentence, some sequences of transitions can be shared across different analyses. This process of sharing computations is implemented with dynamic programming, where transitions are modeled as push computations, which can be tabulated with a deduction system. This allows for polynomial time and space parsing, despite an exponential search space (Kuhlmann et al., 2011).

Each parser state according to the transition system, as defined in Section 2.3, contains two variables (σ, β): the stack σ which contains words seen but not yet completely processed, and the buffer β which represents the current word index. The notation $\sigma|i$ indicates that word i is on top of the stack. The initial state is ([0], 1), and the final state is ([], n).

For this transition system definition, creating a dynamic programming algorithm hinges on modeling the sequence of transitions as a sequence of push computations, as implemented by Kuhlmann et al., 2011 and Cohen et al., 2011. Each push computation can include multiple transition actions, but ultimately results in exactly one node being added to the stack. The most basic push computation is one shift transition, while two push computations can be combined with a reduce (left or right arc) transition to create a new push computation.

$$\text{shift}: [i, j-1] \rightarrow [j-1, j]$$

$$\text{reduce}: [i, k][k, j] \rightarrow [i, j]$$

The dynamic program is modeled as a deduction system (Shieber et al., 1995), where items [i, j] are defined to imply a push computation exists which results in state $(\sigma|i, j)$. With this definition, the action sequence for an entire sentence can be decoded with the Viterbi algorithm, keeping track of the highest scoring index for splitting the push computation into two smaller spanning push computations. The Viterbi path probability $v_n(j)$ at word n for parser state j (with previous parser state i) is calculated:

$$v_n(j) = \max_{i=1}^{N} v_{n-1}(i) \times p_{tr}(j|i) \times p_{gen}(w_n|j) \tag{2.1}$$

The Viterbi path probability for the previous parser state i, where a transition exists from i to j, is multiplied by that transition probability as well as by the probability of generating the next word, given the current parser state j. Although there are N derivations which lead to state j, only the maximum scoring path is recorded as the Viterbi path probability (Jurafsky, 2000).

§ 2.5 Brain Networks for Syntax

The brain regions involved in language comprehension have been widely studied using lesion studies, more invasive analyses during surgery, and relatively newer technologies such as EEG, MEG, and fMRI data. These studies agree that regions involved in language include the inferior frontal gyrus (IFG), the superior temporal gyrus (STG), and parts of the middle temporal gyrus (MTG) and the inferior parietal and angular gyrus in the parietal lobe (Friederici, 2011).

Probing the regions involved in syntax specifically is difficult because setting up an experiment where sentences are manipulated to isolate specific effects, such as syntactic structure difficulty, may result in effects attributable to the specifics of the experimental paradigm. Other data are collected when participants are engaged in natural tasks such as listening to a story.

Interestingly, recent studies using such naturalistic paradigms report brain activation in the right hemisphere in addition to the expected left hemisphere activations (Binder et al., 2009; Huth et al., 2016; Wehbe et al., 2014).

Recent work by Pasquiou et al., 2023 aims to differentiate the regions involved in syntactic and semantic processing from fMRI data while listening to *The Little Prince* by masking either content words or function words from the text (Li et al., 2022). This syntax or semantic specific text was modeled by word embeddings and sequence encodings, which were correlated with fMRI signal. They found overlap in brain regions involved in syntactic and semantic processes, which is expected because of how intertwined the concepts are. For example, compositionality is implicated in both language comprehension processes. However, there were also differences in hemispheres, with the right hemisphere showing greater overlap in function in the same region, while the left hemisphere was more clearly delineated in syntactic and semantic regions.

Crabbé et al., 2019 also studied variable levels of syntactic parallel processing using fMRI the *Little Prince* dataset Li et al., 2022. They found that an RNNG parser with a variable beam width provided complexity metrics, which were predictive of bilateral brain activation in the superior temporal gyrus region.

CHAPTER 3

METHODS

§ 3.1 Parser Training

The construction of the Dependency parser relies on the work of Buys and Blunsom, 2018, which employs the transition based parsing system detailed in Nivre, 2008. However, accuracy is improved by replacing sequence encoding via LSTM with the output of BLOOM, an open-access large language model (Scao et al., 2022). The BigScience Large Open-science Open-access Multilingual Language Model (BLOOM) relies on the transformer architecture introduced by Vaswani et al., 2017, but uses only the decoder. The BLOOM model is trained on the ROOTS corpus, which includes 46 natural languages and 13 programming languages. Of the 1.6TB of training data, 30% is English, and 16% is Chinese. Although the full BLOOM model has 176 billion training parameters, smaller models are also trained using the same architecture and training data, where the smallest model has 560 million trained parameters. This BLOOM-560 model differs from the largest model in that it has 24 layers instead of 70, hidden layers have a width of 1024 instead of 14,336, and there are 16 attention heads instead of 112. The BLOOM-560 model is used throughout this work, since increasing the size to the BLOOM-7b1 model (with 7,069 million parameters) had no significant increase in accuracy. This corresponds to the observations of Oh and Schuler, 2023 and Pasquiou et al., 2023, who correlate large language model encodings to human reading times and fMRI data, respectively, and find that smaller models provide an equal (or better) fit to the human data.

The parser is trained on Universal Dependency corpora, which are available in over 100 languages (De Marneffe et al., 2021). The English model is trained on the English Web Treebank (EWT) corpus, which contains 12,543 training sentences, and 2,001 development set sentences. The Chinese model is trained on the GSD corpus, which contains 3,997 training sentences and 500 development set sentences. Words seen only once in the training data are replaced with unknown word tokens for the purpose of training the next word prediction classifier, but the original word is input to the BLOOM tokenizer for encoding. This tokenizer has a significantly larger vocabulary than the number of words seen in the training data, which is on the order of 3,000 for English, while the BLOOM tokenizer has a vocabulary size of around 250,000 for English. Words not seen even in this larger vocabulary are split into sub-words, potentially down to the character level, resulting in a token sequence, which I interpret by selecting the embedding of the right-most sub-word. This allows sequence encodings to be as detailed as possible, while still limiting the vocabulary during training in order to allow dependency parsing states to impact the word generation probabilities, instead of simply taking the next word prediction probabilities from the BLOOM model.

While training, sentences are shuffled at each epoch, and within batches (batch size = 16), which are created from sentences of the same length. The BLOOM model is pretrained with 560 million parameters, and has 25 layers, with an embedding dimension of 1024. I select the 17th layer as the representation to input into the model, which was decided based on optimization of accuracy on the ParTUT UD corpus, as seen in Fig. 3.1. Instead of fine-tuning the pre-trained BLOOM model on dependency parsing, a Pfeiffer adapter is applied after each layer (Pfeiffer et al., 2020). This bottleneck adapter introduces a new linear layer which reduces the dimension from 1024 down to 64 and back up to 1024, for input into the next pre-trained BLOOM layer. The final output representation (layer=17) is reduced from size 1024 to 650 with a simple single layer feedforward neural network, with a sigmoid nonlinearity, and dropout of 0.5 applied first. During training, gradient norms are clipped to 5.0, the initial learning rate is 1.0, with a decay factor of 1.7 applied every epoch after the initial 6 epochs.

§ 3.1.1 Accuracy

While state of the art methods employ information about the entire sentence (using global methods, or incremental methods with bidirectional embeddings) in order to predict dependency relations, for the purposes of creating a cognitively plausible model, strictly incremental methods must be used without including useful information from words occuring later in the sentence. Accuracy is compared to Buys and Blunsom, 2018 as a "base model" which employs the same transition system with encoding based on an LSTM with random initial weights. This model was retrained on the same UD corpus instead of

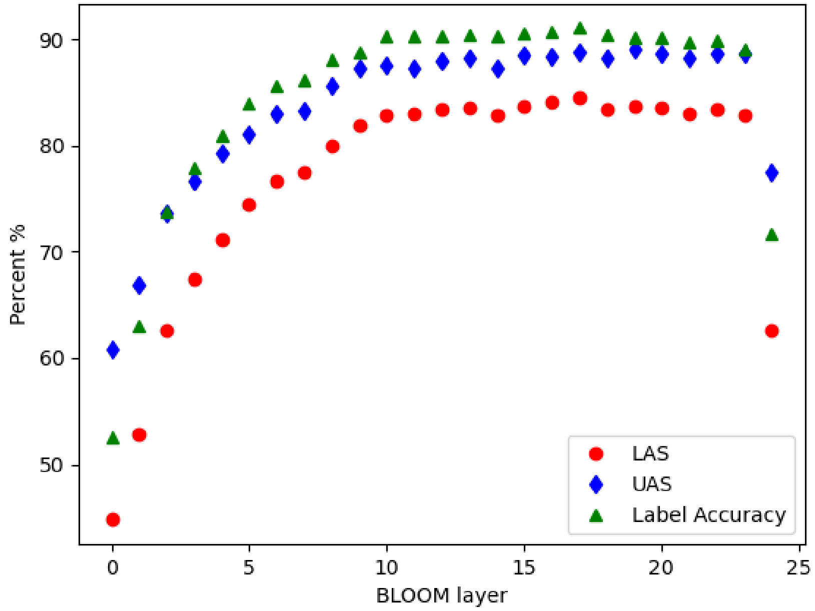

Figure 3.1: Labeled Accuracy Score (LAS), Unlabeled Accuracy Score (UAS), Label Accuracy for Dependency Parser trained and evaluated on UD ParTUT corpus. Each model uses a different BLOOM layer as a sequence encoder.

using the values reported in Buys and Blunsom, 2018 using the Penn TreeBank to make accuracy values directly comparable. Accuracy is improved by initializing with FastText embeddings instead of random intial weights, but training the BLOOM model with the Pfeiffer Adapter provides the greatest increase in accuracy. The improvement in accuracy obtained by including the Pfeiffer Adapter is a significant improvement over using pre-trained BLOOM embeddings directly, which is illustrated in Fig. 3.2.

Table 3.1: Labeled attachment score (LAS), Unlabeled attachment score (UAS), and Label accuracy score for the UD corpus listed

Language	Corpus	Model	LAS	UAS	Label accuracy
English	EWT	base	74.94	80.24	84.74
English	EWT	fastText	75.14	80.62	84.75
English	EWT	BLOOM	80.49	84.74	89.44
Chinese	GSD	base	55.30	62.79	80.87
Chinese	GSD	BLOOM	70.04	75.47	82.55

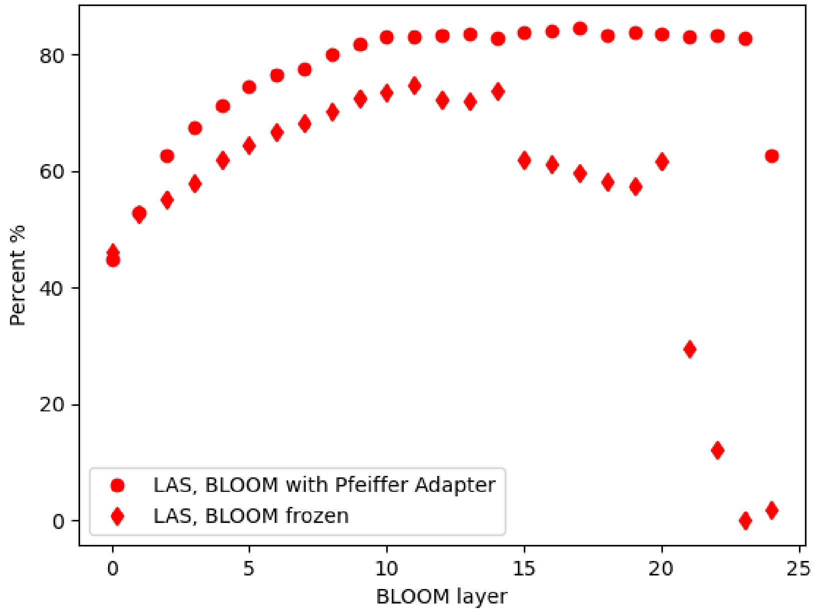

Figure 3.2: Comparison of Labeled Accuracy Score (LAS) for each BLOOM layer encoding as input to the Dependency Parser. Accuracy is improved by adding the Pfeiffer Adapter layers between each BLOOM layer, compared to just BLOOM layers as input to classifiers.

Parser accuracy is evaluated after each epoch. The earliest epoch with highest UAS is chosen to prevent unnecessarily decreasing the loss after accuracy has been reached.

§ 3.2 Derivation Probabilities

The transition system has three possible actions, which can be decomposed into transition and arc direction probabilities, as shown in Table 3.2. The sentence is encoded left-to-right by the BLOOM-560 model, which uses the transformer architecture to create a representation of each word in its linear context, h_i Vaswani et al., 2017. Since the model includes self-attention layers, all words up to and including the current word can influence the vector representation of the current word, to varying degrees according trained weights Jurafsky, 2000. For word w_i, h_i is a vector encoding the word and its left context, $w_{1:i}$. The encoded sequence h predicts transition actions via a binary classifier, and word probabilities via a classifier:

$$p_{tr/dir} = \text{sigmoid}(r^T \text{relu}(W_{ts}h_{\sigma_0} + W_{tb}h_\beta)) \tag{3.1}$$

$$p_{gen} = \text{softmax}(R^T \tanh(W_{gs}h_{\sigma_0} + W_{gb}h_\beta)) \tag{3.2}$$

where $1 = p_{tr}(\text{sh}|h_i, h_j) + p_{tr}(\text{re}|h_i, h_j)$ and $1 = p_{dir}(\text{la}|h_i, h_j) + p_{dir}(\text{ra}|h_i, h_j)$.

Table 3.2: The arc-hybrid transition system, where states (before & after) are indicated by (stack, current index) tuples

Action	Before	After	Arc	Probability
Shift	$(\sigma\|i, j)$	$(\sigma\|i\|j, j+1)$	-	$p_{tr}(\text{sh}\|h_i, h_j)p_{gen}(w_{j+1}\|h_i, h_j)$
Left-arc	$(\sigma\|i, j)$	(σ, j)	$j \rightarrow i$	$p_{tr}(\text{re}\|h_i, h_j)p_{dir}(\text{la}\|h_i, h_j)$
Right-arc	$(\sigma\|l\|i, j)$	$(\sigma\|l, j)$	$l \rightarrow i$	$p_{tr}(\text{re}\|h_i, h_j)p_{dir}(\text{ra}\|h_i, h_j)$

The base model as defined by Buys and Blunsom, 2018, defines the marginal probability of each sentence as the summation over all possible transitions to generate a string of words:

$$p(\mathbf{w_{1:n}}) = \sum_t p(\mathbf{w}_{1:n}, \mathbf{a}_{1:2n}) \tag{3.3}$$

The probability of a potential derivation of a complete sentence is the joint probability $p(\mathbf{w}_{0:n}, \mathbf{a}_{0:2n})$ of parser transitions needed to draw dependency arcs, and the word generation probabilities, which also depend on the arcs already drawn at each word. This joint probability factors as:

$$p(\mathbf{w}_{1:n}, \mathbf{a}_{1:2n}) = \prod_{i=1}^{n} \left(p(w_i|\mathbf{h}_{\sigma_1^{(m_i)}}, \mathbf{h}_{i-1}) \prod_{j=m_i+1}^{m_{i+1}} p(a_j|\mathbf{h}_{\sigma_1^{(j)}}, \mathbf{h}_i) \right) \tag{3.4}$$

where $\mathbf{h}_i$ is the sequence encoding after the string $\mathbf{w_{0:i}}$ has been encoded. Therefore, then encoding represents the word w_i and its left context starting at the ROOT symbol, w_0. The top element of the stack at each transition j is represented by $\sigma_1^{(j)}$, while the variable m_i indicates the number of transitions before w_i is generated. After w_i is generated, the next transition is indexed $m_i + 1$, and this transition is optionally followed by additional *reduce* transitions, leading to the transition indexed m_{i+1} which directly precedes the generation of word w_{i+1}

In order to correlate parser predictions to brain activity, and in particular to take multiple derivations into consideration, it is necessary to include a metric which summarizes the parser prediction information. One such metric is surprisal, revived by Hale, 2001 as a method of predicting sentence-processing difficulty at a word. The basic formula is the logarithm of the inverse of the marginal probability.

$$\log_2\left(\frac{1}{p(y)}\right) = -\log_2(p(y)) \tag{3.5}$$

Extending this to the incremental sentence comprehension situation, the surprisal of a word at position w_n which follows a string ending in w_{n-1}, is a ratio of the information content at the current word minus the previous information content:

$$-\log_2(p(w_{1:n}|w_{1:n-1})) = -\log_2\left(\frac{p(w_n, w_{n-1}, ..., w_1))}{p(w_{n-1}, ..., w_1))}\right) \tag{3.6}$$

$$-\log_2(p(w_{1:n}|w_{1:n-1})) = -\log_2(p(w_{1:n})) + \log_2(p(w_{1:n-1})) \tag{3.7}$$

This value can be calculated from parser transition probabilities by taking the marginal probability of each derivation of interest.

$$p(\mathbf{w_{0:n}}) = \sum_t p(\mathbf{w}_{0:n}, \mathbf{t}_{0:2n}) \tag{3.8}$$

However, since word probabilities are largely based on word frequency and the preceding string context independent of syntax, the word generation probabilities overshadow the differences between $p(\mathbf{w})$ for individual derivations, Fig. 3.3. In order to see differences in competing syntactic derivations, one option is to assume a nongenerative model, and identify each derivation only by the product of its transition probabilities. In addition, surprisal as introduced in Hale, 2001 is actually a sum over all parser states. However, the parser states included in the sum are limited to the top ranking k, which indicates the number of alternate derivations under consideration. This limitation corresponds to the same metric used by Boston et al., 2011 who implemented beam search at various levels of k, while this method avoids

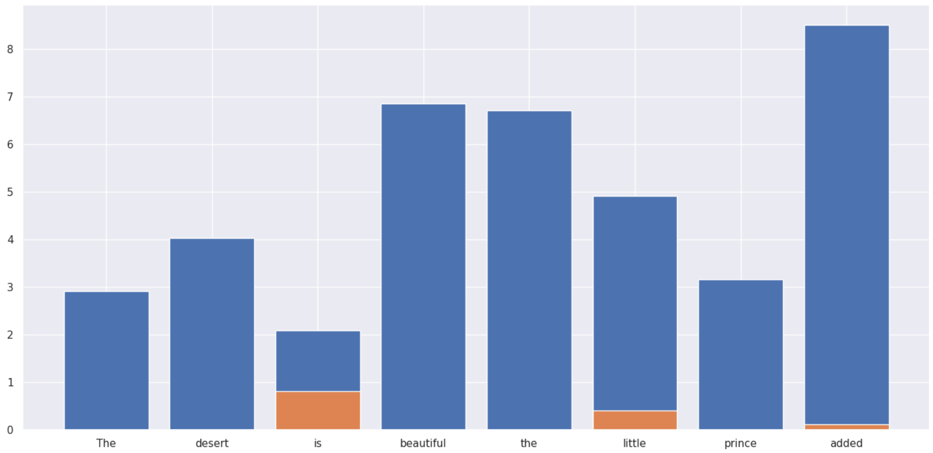

Figure 3.3: A sentence from *The Little Prince*, showing incremental surprisal from overall derivation probability (blue), and syntactic surprisal from transition probabilities only (orange)

decreasing parser accuracy at lower values of k.

$$p(\mathbf{a_n}|\mathbf{a_{n-1}})_k = \sum_{i=1}^{k} p(\mathbf{a_n}|\mathbf{a_{n-1}})_i \tag{3.9}$$

However, the non-generative model suffers from the label bias problem, where global performance suffers when individual actions are trained directly Eisenstein, 2019. Since the probabilities are locally normalized, that is, all possible transitions at each word sum to 1, there is no method of penalizing unlikely word sequences. The inclusion of word generation probabilities remedies this situation, and it may be that the highest scoring transition sequence is not chosen as most likely for sentence based on the word generation probabilities. In an effort to reflect this choice in the generative model without using explicit word generation probabilities, I define the surprisal calculation using conditional probabilities, as the action sequence needed to generate a string of words $p(\mathbf{a}|\mathbf{w})$ where $\mathbf{a}$ is the action sequence given a certain sequence of words.

$$p(\mathbf{a}|\mathbf{w})_k = \sum_{i=1}^{k} p(\mathbf{a}|\mathbf{w})_i \tag{3.10}$$

In the following, I investigate two formulations for $p(a|w)$. The first is a one-step prediction of transitions: $p(a_n|w_{n-1})$. The second is transition sequence only, just as in the non-generative case, but the choice of derivation is "informed" by the highest scoring total probability.

$$p(\mathbf{a}|\mathbf{w})_k = \sum_{i=1}^{k} \frac{p(\mathbf{tr}_{1:2n}, \mathbf{w}_{1:n})_i}{p(\mathbf{w}_{1:n})_i} \tag{3.11}$$

The total number of possible parser states is equal to the sequence of Catalan numbers, so for a sentence of length 1, $C_1 = 1$, while $C_9 = 1430$ which is the average sentence length in the *Little Prince*, while the longest sentence has 52 words, which results in 2.98×10^{28} derivations.

In this way, the parser is complete, and considers all derivation instead of being limited to a beam of the top k options, but this analysis only considers the top ranking k derivations at each word in order to investigate the degree of parallelism in human language comprehension. This is implemented by modifying the Viterbi algorithm to keep a list of the top k transition actions at each split-point, instead of only keeping track of the top 1 scoring action.

§ 3.3 fMRI Datasets

The English data were collected and preprocessed by John Hale, Shohini Bhattasali, and Jixing Li with the help of Nathan Spreng and Wen-Ming Lu. The Chinese data were collected and preprocessed by Jixing Li with the help of Yiming Yang.

§ 3.3.1 Participants

All study participants were healthy, right-handed, young adults. They self-identified as native speakers of the language in question, and had no history of psychiatric, neurological or other medical illness that could compromise cognitive functions. All participants were paid and gave written informed consent prior to participation, in accordance with the guidelines of the Human Research Participant Protection Program at Cornell University or the Ethics Committee at Jiangsu Normal University for English and Chinese particpants, respectively. The English dataset includes 51 participants (32 female, mean age = 21.3, range = 18-37), and the Chinese dataset includes 35 participants (15 female, mean age = 19.3, range = 18-25).

§ 3.3.2 Data Acquisition

The English audio stimulus is an English translation of *The Little Prince*, read by Karen Savage. The Chinese audio stimulus is a Chinese translation of *The Little Prince*, read by a professional female Chinese broadcaster. The English and Chinese audiobooks are 94 and 99 minutes in length, respectively. The presentations were divided into nine sections, each lasting around ten minutes. Participants listened passively to the nine sections and completed four quiz questions after each section (36 questions in total). These questions were used to confirm participant comprehension of the story.

The English and Chinese brain imaging data were acquired with a 3T MRI GE Discovery MR750 scanner with a 32-channel head coil. Anatomical scans were acquired using a T1-weighted volumetric magnetization prepared rapid gradient-echo pulse sequence. Blood-oxygen-level-dependent (BOLD) functional scans were acquired using a multi-echo planar imaging sequence with online reconstruction (TR = 2000 ms; TE's = 12.8, 27.5, 43 ms; F = 77○ 200 ; matrix size = 72 x 72; FOV = 240.0 mm x 240.0 mm; 2 x image acceleration; 33 axial slices, voxel size = 3.75 x 3.75 x 3.8 mm).

§ 3.3.3 Data Preprocessing

The English and Chinese fMRI data were preprocessed using AFNI version 16 (Cox, 1996). The first 4 volumes in each run were excluded from analyses to allow for T1-equilibration effects. Multi-echo independent components analysis (ME-ICA, Kundu et al., 2012) was used to denoise data for motion, physiology, and scanner artifacts. Images were then spatially normalized to the standard space of the Montreal Neurological Institute (MNI) atlas, yielding a volumetric time series resampled at 2 mm cubic voxels.

§ 3.4 Statistical Analysis

The goal of the analysis is to compare a model of fMRI signal from surprisal at varying levels of k. For each subject, the BOLD (Blood Oxygenation Level Dependent) signal is modeled by a GLM at each voxel. The regressors in the GLM include word rate, fundamental frequency (f0), word frequency, root mean square intensity (RMS), and surprisal at the top k derivations. Root Mean Square intensity (RMS) an indicator of audio intensity at every 10 ms of the audio; word rate is a stick function marking the offset of each spoken word; frequency is derived from the individual words in a movie subtitles database (Brysbaert and New, 2009). Such predictors are here to ensure that conclusions about parsing difficulty would be specific to parsing instead of general speech comprehension, which is known to be affected by these factors. Specifically, lexical frequency was added as a covariate of noninterest, to statistically factor out effects of general word frequency that would correlate with the overal expectation at each word.

Surprisal for each word was aligned to the offset of each word in the audiobook. Model comparisons using cross-validated coefficient of determination (r^2) maps were carried out in order to evaluate the goodness of fit of the two surprisal calculations with BOLD signal. The predictors were convolved using SPM's (Friston et al., 2007) canonical HRF (Hemodynamic Response Function).

For every participant, the goal is to compute how much the inclusion of surprisal at varying levels of k increases the cross-validated r^2, from a baseline model including only the control variables. Therefore, the r^2 scores represent the variance explained in each voxel by the addition of surprisal to the model as a predictor. Surprisal regressors at various levels of k are added to the base model separately in order keep the number of parameters in each linear regression model constant.

To compare the impact of the number of alternate derivations included in the surprisal calculation on fMRI signal explanation (i.e. r^2 increase of each variable) a paired t-test was performed on each individual r^2 map, and obtained z-maps showing the regions where surprisal at one level of parallelism explains the signal significantly better than surprisal at the other.

CHAPTER 4

RESULTS AND DISCUSSION

§ 4.1 Results

§ 4.1.1 Complexity Metrics

The results for the English Universal Dependencies model are compared across different complexity metrics. The non-generative model selects for the top k transition probabilities, while the *prediction* model makes a 1-step prediction from the joint transition and word probability at $n - 1$ to the next transition only at word n.

In the following z-maps, the threshold K indicates the minimum voxel cluster size, while k indicates the number of parser states considered by the complexity metric.

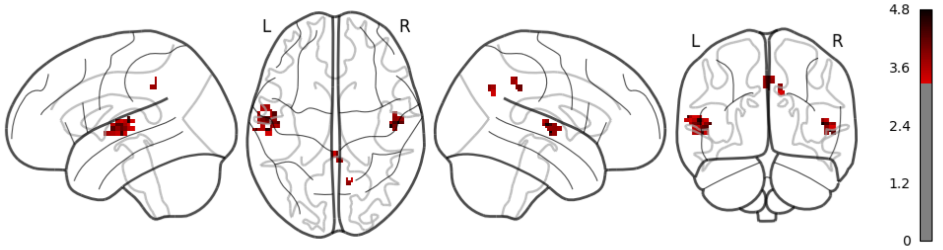

Figure 4.1: English Brain z-maps showing the significant clusters (K>25; p < .001 uncorrected) for the UD model comparison between surprisal at k=1 vs k=5 for parser transitions only.

Both English paired t-tests on r^2 maps show mainly bilateral activation of the superior temporal gyrus. This region is indicated in other studies of syntactic ambiguity, e.g. Dunagan et al., 2022. This study analyzes Object-Extracted Relative Clauses (ORC), and finds superior temporal gyrus activation in Chinese ORC processing, and suggests, in accord with Jäger et al., 2015 that syntactic ambiguity is involved in the comprehension of Chinese ORCs.

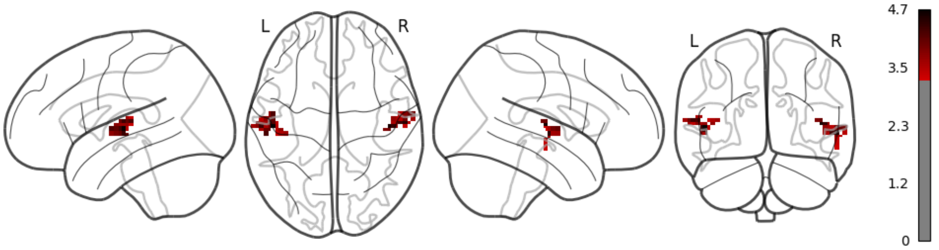

Figure 4.2: English Brain z-maps showing the significant clusters (K>25; p < .001 uncorrected) for the UD model comparison between surprisal at k=1 vs k=5 for $p(a_n|w_{n-1})$.

The final complexity metric calculates the probability of a derivation as the product of transition probabilities which is the same as the non-generative model, but chooses the top k informed by the generative model.

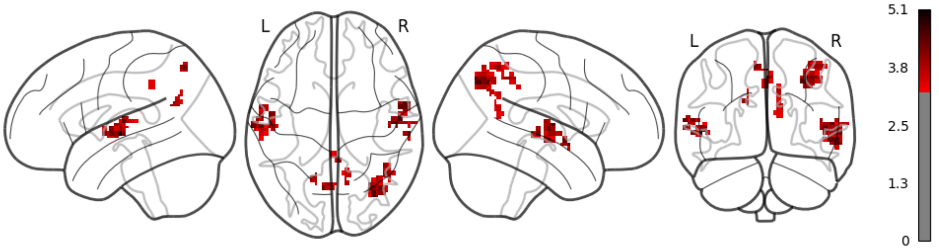

Figure 4.3: English Brain z-maps showing the significant clusters (K>25; p < .001 uncorrected) for the UD model comparison between surprisal at k=1 vs k=5 for transition probabilities chosen by maximum scoring generative probabilities.

This *informed* result differs from the other metrics in that it emphasizes the penalty incurred by reanalysis - when the top derivation changes from one word to the next. With the other metrics, the top derivation can only be abandoned when it decreases in probability, while in practice incorrect transitions do not necessarily have low probabilities, and avoiding ungrammatical derivations is the responsibility of word generation probabilities. Since the *informed* metric uses the full word generation information, surprisal is increased over that of the *prediction* method when reanalysis occurs. This may explain the extra activations at the right temporo-parietal junction.

Pasquiou et al., 2023 compared brain regions between syntax and semantics, and found that the right temporo-parietal junction shows strong overlap with syntax and semantics, while the corresponding left region is more dominated by semantic processing. This corresponds to Fig. 4.3 which also shows only relatively small regions of the temporo-parietal junction in the left hemisphere, and strong activation in the right hemisphere.

Wehbe et al., 2014 also correlated various linguistic features with fMRI data in a naturalistic comprehension study. The syntactic features studies were part of speech, ordinal position in the sentence, and dependency role in the sentence. These syntactic features predicted bilateral brain activation at the temporo-parietal junction, but did not predict the STG activations observed in this study. This shows that modeling dependency relations beyond the role label can give extra information about syntactic processing which is key to understanding the brain regions involved.

§ 4.1.2 SUD Results

The results from the English Surface-Syntactic Universal Dependencies (SUD) model generally coincide with those from the UD model. However, the regions involved are limited to the superior temporal gyrus. Since the SUD model relies on function words instead of content words to head phrases, the regions also involved in semantic processing may be less activated.

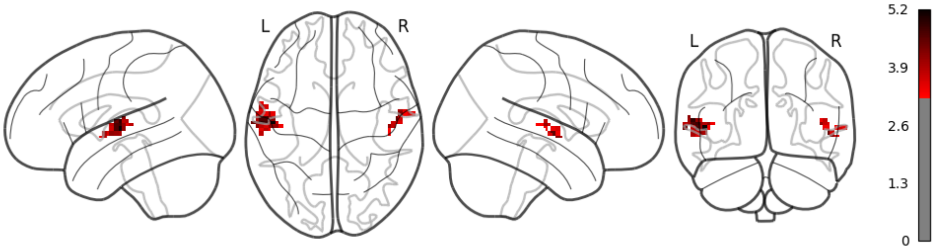

Figure 4.4: English Brain z-maps showing the significant clusters (K>25; p < .001 uncorrected) for the SUD model comparison between surprisal at k=1 vs k=5 for parser transitions only.

§ 4.1.3 Chinese Results

The Chinese UD model results also coincide with their English counterparts. The right hemisphere seems to have slightly stronger activation in Chinese, while for English the left hemisphere is generally slightly stronger.

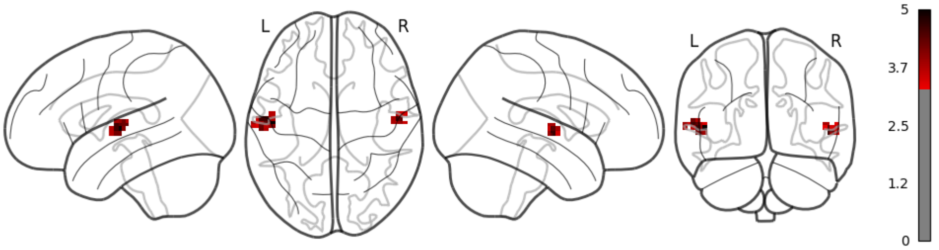

Figure 4.5: English Brain z-maps showing the significant clusters (K>25; p < .001 uncorrected) for the SUD model comparison between surprisal at k=1 vs k=5 for parser transitions chosen by maximum scoring generative probabilities.

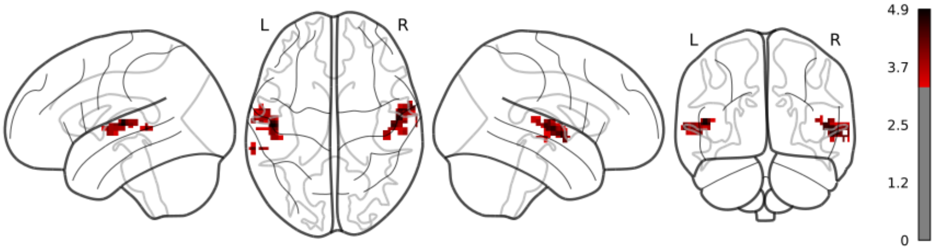

Figure 4.6: Chinese Brain z-maps showing the significant clusters (K>25; p < .001 uncorrected) for the UD model comparison between surprisal at k=1 vs k=5 for parser transitions only.

§ 4.2 Discussion

Both Chinese and English paired t-tests on r^2 maps show mainly bilateral activation of the superior temporal gyrus. This region is expected based on other studies of syntactic processing, such as Crabbé et al., 2019, Pasquiou et al., 2023 and Dunagan et al., 2022.

Although language processing has classically been viewed as taking place in the left hemisphere for right handed individuals, Mason et al., 2003 have found that right-hemisphere regions are activated when syntactic ambiguity beyond a certain threshold is encountered. This bilateral activation supports the theory that ambiguity resolution does take place in naturalistic speech comprehension, beyond specific laboratory situations such as garden-path sentences. Mason et al., 2003 found higher levels of brain activations for ambiguous sentences even when ambiguities were resolved in favor of expected constructions, which corresponds to the theory of parallel processing.

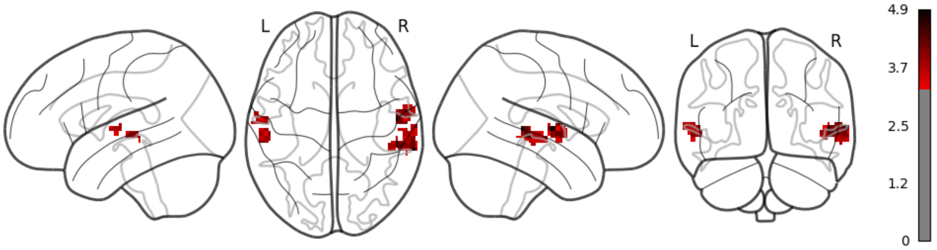

Figure 4.7: Chinese Brain z-maps showing the significant clusters (K>25; p < .001 uncorrected) for the model comparison between surprisal at k=1 vs k=2.

The improvement of the model at $k = 5$ also indicates support for the theory that human sentence processing is largely parallel instead of serial. Although many words in both models have surprisal approaching zero, indicating a non-ambiguous situation, the difference among the models occurs with certain words where surprisal at $k = 1$ predicts difficulty, while surprisal at $k = 5$ predicts no difficulty because the virtually all the probability mass is contained within the top 5 derivations. Since surprisal at $k = 5$ provides a greater r^2 increase, this provides evidence that several alternate derivations are actually being updated during human sentence comprehension.

CHAPTER 5

CONCLUSION

Using an incremental, generative, dependency parser enhanced with sequence encodings from a large language model, several "syntactic surprisal" measures were proposed in order to analyze the syntactic ambiguity present when listening to an audiobook. These metrics were calculated for partial dependency trees, which provided a measure of processing difficulty at each word in the text. It was possible to correlate these metrics with BOLD fMRI signal, validating the hypothesis that derivations with low predicted probability require greater effort to understand.

The surprisal metrics were validated at various levels of parallel processing. Since increasing k provides a significantly better predictor for the data, there is new evidence in favor of parallel processing.

Using these metrics also allowed for identification of the brain regions associated with increased activation for higher levels of k across English and Chinese, namely the superior temporal gyrus in both left and right hemispheres. Additional activation at the temporo-parietal junction was also observed for the surprisal metric which most emphasize probability change due to reanalysis. This evidence shows that these regions are involved in disambiguating among competing syntactic derivations. The bilateral activation coincides with the expectation to see greater right hemisphere activation in naturalistic listening data compared with studies using individual sentences, as well as with the prediction that the right hemisphere regions are recruited when processing difficulty exceeds a threshold.

The brain regions involved in syntactic disambiguation were found to correspond between English and Chinese data. Future work may validate these results across additional languages, particularly where greater typological differences exist, especially variation in word order, morphology, and syntactic headedness. Future work may also categorize temporary ambiguity according to different criteria, such as various dependency parsing transition systems and annotation schemas, although the current results predict the same brain regions to be implicated.

BIBLIOGRAPHY

Binder, J. R., Desai, R. H., Graves, W. W., & Conant, L. L. (2009). Where is the semantic system? a critical review and meta-analysis of 120 functional neuroimaging studies. *Cerebral cortex*, *19*(12), 2767–2796.

Boston, M. F., Hale, J. T., Vasishth, S., & Kliegl, R. (2011). Parallel processing and sentence comprehension difficulty. *Language and Cognitive Processes*, *26*(3), 301–349.

Buys, J., & Blunsom, P. (2018). Neural syntactic generative models with exact marginalization. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 942–952.

Caucheteux, C., & King, J.-R. (2022). Brains and algorithms partially converge in natural language processing. *Nature Communications Biology*, *5*(1), 134. https://doi.org/10.1038/s42003-022-03036-1

Cohen, S. B., Gómez-Rodríguez, C., & Satta, G. (2011). Exact inference for generative probabilistic non-projective dependency parsing. *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, 1234–1245.

Covington, M. A. (2001). A fundamental algorithm for dependency parsing. *Proceedings of the 39th annual ACM southeast conference*, *1*.

Cowan, N. (2001). The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and brain sciences*, *24*(1), 87–114.

Crabbé, B., Fabre, M., & Pallier, C. (2019). Variable beam search for generative neural parsing and its relevance for the analysis of neuro-imaging signal. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 1150–1160.

De Marneffe, M.-C., Manning, C. D., Nivre, J., & Zeman, D. (2021). Universal dependencies. *Computational linguistics*, *47*(2), 255–308.

De Marneffe, M.-C., & Nivre, J. (2019). Dependency grammar. *Annual Review of Linguistics*, *5*, 197–218.

DeLong, K. A., Troyer, M., & Kutas, M. (2014). Pre-processing in sentence comprehension: Sensitivity to likely upcoming meaning and structure. *Language and linguistics compass*, *8*(12), 631–645.

Dunagan, D., Stanojević, M., Coavoux, M., Zhang, S., Bhattasali, S., Li, J., Brennan, J., & Hale, J. (2022). Neural correlates of object-extracted relative clause processing across english and chinese. *Neurobiology of Language*, 1–43.

Dyer, C., Kuncoro, A., Ballesteros, M., & Smith, N. (2016). Recurrent neural network grammars. *In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 199–209.

Eisenstein, J. (2019). *Introduction to natural language processing*. MIT press.

Fodor, J., Bever, A., Garrett, T., et al. (1974). The psychology of language: An introduction to psycholinguistics and generative grammar.

Friederici, A. D. (2011). The brain basis of language processing: From structure to function. *Physiological reviews*, *91*(4), 1357–1392.

Futrell, R., Mahowald, K., & Gibson, E. (2015). Large-scale evidence of dependency length minimization in 37 languages. *Proceedings of the National Academy of Sciences*, *112*(33), 10336–10341.

Gerdes, K., Guillaume, B., Kahane, S., & Perrier, G. (2021). Starting a new treebank? go SUD! *Proceedings of the Sixth International Conference on Dependency Linguistics (Depling, SyntaxFest 2021)*, 35–46.

Hahn, M., & Xu, Y. (2022). Crosslinguistic word order variation reflects evolutionary pressures of dependency and information locality. *Proceedings of the National Academy of Sciences*, *119*(24), e2122604119.

Hale, J. T. (2001). A probabilistic Earley parser as a psycholinguistic model. *Second meeting of the North American chapter of the association for computational linguistics*.

Hale, J. T., Campanelli, L., Li, J., Bhattasali, S., Pallier, C., & Brennan, J. R. (2022). Neurocomputational models of language processing. *Annual Review of Linguistics*, *8*, 427–446.

Hale, J. T., Dyer, C., Kuncoro, A., & Brennan, J. (2018). Finding syntax in human encephalography with beam search. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2727–2736. https://doi.org/10.18653/v1/P18-1254

Hudson, R. (1990). English word grammar.

Huth, A. G., De Heer, W. A., Griffiths, T. L., Theunissen, F. E., & Gallant, J. L. (2016). Natural speech reveals the semantic maps that tile human cerebral cortex. *Nature*, *532*(7600), 453–458.

Jäger, L., Chen, Z., Li, Q., Lin, C.-J. C., & Vasishth, S. (2015). The subject-relative advantage in chinese: Evidence for expectation-based processing. *Journal of Memory and Language*, *79*, 97–120.

Jurafsky, D. (2000). *Speech & language processing*. Pearson Education India.

Jurafsky, D. (1996). A probabilistic model of lexical and syntactic access and disambiguation. *Cognitive science*, *20*(2), 137–194.

Kamide, Y., Altmann, G. T., & Haywood, S. L. (2003). The time-course of prediction in incremental sentence processing: Evidence from anticipatory eye movements. *Journal of Memory and Language*, *49*, 133–156.

Kudo, T., & Matsumoto, Y. (2002). Japanese dependency analysis using cascaded chunking. *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*.

Kuhlmann, M., Gómez-Rodríguez, C., & Satta, G. (2011). Dynamic programming algorithms for transition -based dependency parsers. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 673–682. https://aclanthology.org/P11-1068

Li, J., Bhattasali, S., Zhang, S., Franzluebbers, B., Luh, W.-M., Spreng, R. N., Brennan, J. R., Yang, Y., Pallier, C., & Hale, J. (2022). Le Petit Prince multilingual naturalistic fmri corpus. *Scientific data*, *9*(1), 530.

Liu, H. (2008). Dependency distance as a metric of language comprehension difficulty. *Journal of Cognitive Science*, *9*(2), 159–191.

Marslen-Wilson, W. (1973). Linguistic structure and speech shadowing at very short latencies. *Nature*, *244*, 522–523.

Mason, R. A., Just, M. A., Keller, T. A., & Carpenter, P. A. (2003). Ambiguity in the brain: What brain imaging reveals about the processing of syntactically ambiguous sentences. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *29*(6), 1319.

Mel'cuk, I. A., et al. (1988). *Dependency syntax: Theory and practice*. SUNY press.

Nivre, J. (2003). An efficient algorithm for projective dependency parsing. *Proceedings of the eighth international conference on parsing technologies*, 149–160.

Nivre, J. (2004). Incrementality in deterministic dependency parsing. *Proceedings of the workshop on incremental parsing: Bringing engineering and cognition together*, 50–57.

Nivre, J. (2008). Algorithms for deterministic incremental dependency parsing. *Computational Linguistics*, *34*(4), 513–553.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryigit, G., Kübler, S., Marinov, S., & Marsi, E. (2007). Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, *13*(2), 95–135.

Oh, B.-D., & Schuler, W. (2023). 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.

Pasquiou, A., Lakretz, Y., Thirion, B., & Pallier, C. (2023). Information-restricted neural language models reveal different brain regions' sensitivity to semantics, syntax and context. *arXiv preprint arXiv:2302.14389*.

Pfeiffer, J., Rücklé, A., Poth, C., Kamath, A., Vulić, I., Ruder, S., Cho, K., & Gurevych, I. (2020). Adapterhub: A framework for adapting transformers. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020): Systems Demonstrations*, 46–54. https://www.aclweb.org/anthology/2020.emnlp-demos.7

Scao, T. L., Fan, A., Akiki, C., Pavlick, E., Ilić, S., Hesslow, D., Castagné, R., Luccioni, A. S., Yvon, F., Gallé, M., et al. (2022). Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Schrimpf, M., Blank, I. A., Tuckute, G., Kauf, C., Hosseini, E. A., Kanwisher, N., Tenenbaum, J. B., & Fedorenko, E. (2021). The neural architecture of language: Integrative modeling converges on predictive processing. *Proceedings of the National Academy of Sciences*, *118*(45), e2105646118. https://doi.org/10/gncxrj

Shieber, S. M., Schabes, Y., & Pereira, F. C. (1995). Principles and implementation of deductive parsing. *The Journal of logic programming*, *24*(1-2), 3–36.

Tanenhaus, M., Spivey-Knowlton, M., Eberhard, K., & Sedivy, J. (1995). Integration of visual and linguistic information in spoken language comprehension. *Science*, *268*, 1632–1634.

Tesnière, L. (1959). Elements de syntaxe structurale.

Van Gompel, R. P., Pickering, M. J., Pearson, J., & Liversedge, S. P. (2005). Evidence against competition during syntactic ambiguity resolution. *Journal of Memory and Language*, *52*(2), 284–307.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, *30*.

Wehbe, L., Murphy, B., Talukdar, P., Fyshe, A., Ramdas, A., & Mitchell, T. (2014). Simultaneously uncovering the patterns of brain regions involved in different story reading subprocesses. *PloS one*, *9*(11), e112575.

Yamada, H., & Matsumoto, Y. (2003). Statistical dependency analysis with support vector machines. *Proceedings of the eighth international conference on parsing technologies*, 195–206.

Zwicky, A. M. (1985). Heads. *Journal of linguistics*, *21*(1), 1–29.