An Analysis of Frequency- and Memory-Based Processing Costs

Marten van Schijndel
The Ohio State University
vanschm@ling.osu.edu

William Schuler
The Ohio State University
schuler@ling.osu.edu

Abstract

The frequency of words and syntactic constructions has been observed to have a substantial effect on language processing. This begs the question of what causes certain constructions to be more or less frequent. A theory of grounding (Phillips, 2010) would suggest that cognitive limitations might cause languages to develop frequent constructions in such a way as to avoid processing costs. This paper studies how current theories of working memory fit into theories of language processing and what influence memory limitations may have over reading times. Measures of such limitations are evaluated on eye-tracking data and the results are compared with predictions made by different theories of processing.

1 Introduction

Frequency effects in language have been isolated and observed in many studies (Trueswell, 1996; Jurafsky, 1996; Hale, 2001; Demberg and Keller, 2008). These effects are important because they illuminate the ontogeny of language (how individual speakers have acquired language), but they do not answer questions about the phylogeny of language (how the language came to its current form).

Phillips (2010) has hypothesized that grammar rule probabilities may be grounded in memory limitations. Increased delays in processing center-embedded sentences as the number of embeddings increases, for example, are often explained in terms of a complexity cost associated with maintaining incomplete dependencies in working memory (Gibson, 2000; Lewis and Vasishth, 2005). Other studies have shown a link between processing delays and the low frequency of center-embedded constructions like object relatives (Hale, 2001), but they have not explored the source of this low frequency. A grounding hypothesis would claim that the low probability of generating such a structure may arise from an associated memory load. In this account, while these complexity costs may involve language-specific concepts such as referent or argument linking, the underlying explanation would be one of memory limitations (Gibson, 2000) or neural activation (Lewis and Vasishth, 2005).

This paper seeks to explore the different predictions made by these theories on a broad-coverage corpus of eye-tracking data (Kennedy et al., 2003). In addition, the current experiment seeks to isolate memory effects from frequency effects in the same task. The results show that memory load measures are a significant factor even when frequency measures are residualized out.

The remainder of this paper is organized as follows: Sections 2 and 3 describe several frequency and memory measures. Section 4 describes a probabilistic hierarchic sequence model that allows all of these measures to be directly computed. Section 5 describes how these measures were used to predict reading time durations on the Dundee eye-tracking corpus. Sections 6 and 7 present results and discuss.

2 Frequency Measures

2.1 Surprisal

One of the strongest predictors of processing complexity is surprisal (Hale, 2001). It has been shown in numerous studies to have a strong correlation with reading time durations in eye-tracking and self-paced reading studies when calculated with a variety

of models (Levy, 2008; Roark et al., 2009; Wu et al., 2010).

Surprisal predicts the integration difficulty that a word x_t at time step t presents given the preceding context and is calculated as follows:

$$surprisal(x_t) = -\log_2 \left(\frac{\sum_{s \in S(x_1 \ldots x_t)} P(s)}{\sum_{s \in S(x_1 \ldots x_{t-1})} P(s)} \right) \quad (1)$$

where $S(x_1 \ldots x_t)$ is the set of syntactic trees whose leaves have $x_1 \ldots x_t$ as a prefix.[1]

In essence, surprisal measures how unexpected constructions are in a given context. What it does not provide is an explanation for why certain constructions would be less common and thus more surprising.

2.2 Entropy Reduction

Processing difficulty can also be measured in terms of entropy (Shannon, 1948). A larger entropy over a random variable corresponds to greater uncertainty over the observed value it will take. The entropy of a syntactic derivation over the sequence $x_1 \ldots x_t$ is calculated as:[2]

$$H(x_{1\ldots t}) = \sum_{s \in S(x_1 \ldots x_t)} -P(s) \cdot \log_2 P(s) \quad (2)$$

Reduction in entropy has been found to predict processing complexity (Hale, 2003; Hale, 2006; Roark et al., 2009; Wu et al., 2010; Hale, 2011):

$$\Delta H(x_{1\ldots t}) = \max(0, H(x_{1\ldots t-1}) - H(x_{1\ldots t})) \quad (3)$$

This measures the change in uncertainty about the discourse as each new word is processed.

3 Memory Measures

3.1 Dependency Locality

In Dependency Locality Theory (DLT) (Gibson, 2000), complexity arises from intervening referents introduced between a predicate and its argument. Under the original formulation of DLT, there is a *storage* cost for each new referent introduced and an *integration* cost for each referent intervening in a dependency projection. This is a simplification made for ease of computation, and subsequent work has found DLT to be more accurate cross-linguistically if the intervening elements are structurally defined rather than defined in terms of referents (Kwon et al., 2010). That is, simply having a particular referent intervene in a dependency projection may not have as great an effect on processing complexity as the syntactic construction the referent appears in. Therefore, this work reinterprets the costs of dependency locality to be related to the events of beginning a center embedding (storage) and completing a center embedding (integration). Note that anti-locality effects (where longer dependencies are easier to process) have also been observed in some languages, and DLT is unable to account for these phenomena (Vasishth and Lewis, 2006).

3.2 ACT-R

Processing complexity has also been attributed to confusability (Lewis and Vasishth, 2005) as defined in domain-general cognitive models like ACT-R (Anderson et al., 2004).

ACT-R is based on theories of neural activation. Each new word is *encoded* and stored in working memory until it is *retrieved* at a later point for modification before being re-encoded into the parse. A newly observed sign (word) associatively activates any appropriate arguments from working memory, so multiple similarly appropriate arguments would slow processing as the parser must choose between the highly activated hypotheses. Any intervening signs (words or phrases) that modify a previously encoded sign re-activate it and raise its resting activation potential. This can ease later retrieval of that sign in what is termed an *anti-locality* effect, contra predictions of DLT. In this way, returning out of an embedded clause can actually speed processing by having primed the retrieved sign before it was needed. ACT-R attributes locality phenomena to frequency effects (e.g. unusual constructions) overriding such priming and to activation decay if embedded signs do not prime the target sign through modification (as in parentheticals). Finally, ACT-R predicts something like DLT's storage cost due to the need to differentiate each newly encoded sign from

[1] The parser in this study uses a beam. However, given high parser accuracy, Roark (2001) showed that calculating complexity metrics over a beam should obtain similar results to the full complexity calculation.

[2] The incremental formulation used here was first proposed in Wu et al. (2010).

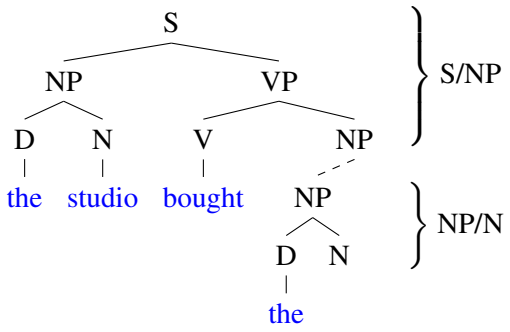

Figure 1: Two disjoint connected components of a phrase structure tree for the sentence *The studio bought the publisher's rights,* shown immediately prior to the word *publisher*.

those previously encoded (similarity-based encoding interference) (Lewis et al., 2006).

3.3 Hierarchic Sequential Prediction

Current models of working memory in structured tasks are defined in terms of hierarchies of sequential processes, in which superordinate sequences can be interrupted by subordinate sequences and resume when the subordinate sequences have concluded (Botvinick, 2007). These models rely on *temporal* cueing as well as content-based cueing to explain how an interrupted sequence may be recalled for continuation.

Temporal cueing is based on a context of temporal features for the current state (Howard and Kahana, 2002). The temporal context in which the subordinate sequence concludes must be similar enough to the temporal context in which it was initiated to recall where in the superordinate sequence the subordinate sequence occurred. For example, the act of making breakfast may be interrupted by a phone call. Once the call is complete, the temporal context is sufficiently similar to when the call began that one is able to continue preparing breakfast. The association between the current temporal context and the temporal context prior to the interruption is strong enough to cue the next action.

Temporal cueing is complemented by *sequential* (content-based) cueing (Botvinick, 2007) in which the content of an individual element is associated with, and thus cues, the following element. For example, recalling the 20th note of a song is difficult, but when playing the song, each note cues the following note, leading one to play the 20th note without difficulty.

Hierarchic sequential prediction may be directly applicable to processing syntactic center embeddings (van Schijndel et al., in press). An ongoing parse may be viewed graph-theoretically as one or more connected components of incomplete phrase structure trees (see Figure 1). Beginning a new subordinate sequence (a center embedding) introduces a new connected component, disjoint from that of the superordinate sequence. As the subordinate sequence proceeds, the new component gains associated discourse referents, each sequentially cued from the last, until finally it merges with the superordinate connected component at the end of the embedded clause, forming a single connected component representing the parse up to that point. Since it is not connected to the subordinate connected component prior to merging, the superordinate connected component must be recalled through temporal cueing.

McElree (2001; 2006) has found that retrieval of any non-focused (or in this case, unconnected) element from memory leads to slower processing. Therefore, integrating two disjoint connected components should be expected to incur a processing cost due to the need to recall the current state of the superordinate sequence to continue the parse. Such a cost would corroborate a DLT-like theory where integration slows processing.

3.4 Dynamic Recruitment of Additional Processing Resources

Language processing is typically centered in the left hemisphere of the brain (for right-handed individuals). Just and Varma (2007) provide fMRI results suggesting readers dynamically recruit additional processing resources such as the right-side homologues of the language processing areas of the brain when processing center-embedded constructions. Once an embedded construction terminates, the reader may still have temporary access to these extra processing resources, which may briefly speed processing.

This hypothesis would, therefore, predict an *encoding* cost when a center embedding is initiated. The resulting inhibition would trigger recruitment of additional processing resources, which would then

allow the rest of the embedded structure to be processed at the usual speed. Upon completing an embedding, the difficulty arising from memory retrieval (McElree, 2001) would be ameliorated by these extra processing resources, and the reduced processing complexity arising from reduced memory load would yield a temporary *facilitation* in processing. No longer requiring the additional resources to cope with the increased embedding, the processor would release them, returning the processor to its usual speed. Unlike anti-locality, where processing is facilitated in longer passages due to accumulating probabilistic evidence, a model of dynamic recruitment of additional processing resources would predict universal facilitation after a center embedding of any length, modulo frequency effects.

3.5 Embedding Difference

Wu et al. (2010) propose an explicit measure of the difficulty associated with processing center-embedded constructions, which is similar to the predictions of dynamic recruitment and is defined in terms of changes in memory load. They calculate a probabilistically-weighted average embedding depth as follows:

$$\mu_{emb}(x_1 \ldots x_t) = \sum_{s \in S(x_1 \ldots x_t)} d(s) \cdot P(s) \qquad (4)$$

where $d(s)$ returns the embedding depth of the derivation s at x_t in a variant of a left-corner parsing process.[3] Embedding difference may then be derived as:

$$\begin{aligned} \mathit{EmbDiff}(x_1 \ldots x_t) = & \mu_{emb}(x_1 \ldots x_t) - \\ & \mu_{emb}(x_1 \ldots x_{t-1}) \end{aligned} \qquad (5)$$

This is hypothesized to correlate positively with processing load: increasing the embedding depth increases processing load and decreasing it reduces processing load. Note that embedding difference makes the opposite prediction from DLT in that integrating an embedded clause is predicted to speed processing. In fact, the predictions of embedding difference are such that it may be viewed as an implementation of the predictions of a hierarchic sequential processing model with dynamic recruitment of additional resources.

[3] As pointed out by Wu et al. (2010), in practice this can be computed over a beam of potential parses in which case it must be normalized by the total probability of the beam.

4 Model

This paper uses a hierarchic sequence model implementation of a left-corner parser variant (van Schijndel et al., in press), which represents connected components of phrase structure trees in hierarchies of hidden random variables. This requires, at each time step t:

- a hierarchically-organized set of N connected component states q_t^n, each consisting of an active sign of category $a_{q_t^n}$, and an awaited sign of category $b_{q_t^n}$, separated by a slash '/'; and
- an observed word x_t.

Each connected component state in this model then represents a contiguous portion of a phrase structure tree (see Figure 1 on preceding page).

The operations of this parser can be defined as a deductive system (Shieber et al., 1995) with an input sequence consisting of a top-level connected component state $\top/\top$, corresponding to an existing discourse context, followed by a sequence of observed words $x_1, x_2, \ldots$[4] If an observation x_t can attach as the awaited sign of the most recent (most subordinate) connected component a/b, it is hypothesized to do so, turning this incomplete sign into a complete sign a (F–, below); or if the observation can serve as a lower descendant of this awaited sign, it is hypothesized to form the first complete sign a' in a newly initiated connected component (F+):

$$\frac{\begin{array}{cc} a/b & x_t \end{array}}{a}\, b \rightarrow x_t \qquad \text{(F–)}$$

$$\frac{\begin{array}{cc} a/b & x_t \\ a/b & a' \end{array}}{}\, b \stackrel{+}{\rightarrow} a' \ldots ; \;\; a' \rightarrow x_t \qquad \text{(F+)}$$

Then, if either of these complete signs (a or a' above, matched to a'' below) can attach as an initial

[4] A deductive system consists of inferences or productions of the form: $\frac{P}{Q} R$, meaning premise P entails conclusion Q according to rule R.

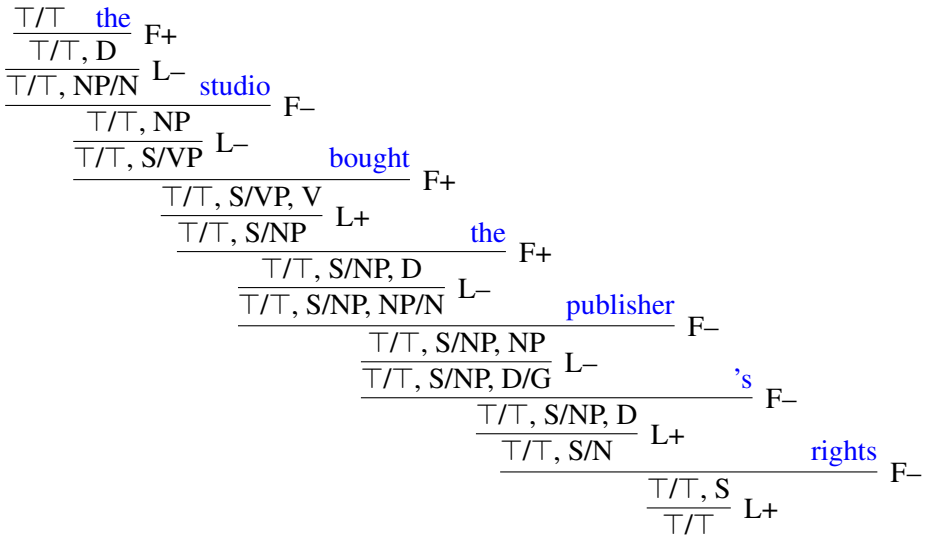

Figure 2: Example parse (in the form of a deductive proof) of the sentence *The studio bought the publisher's rights,* using F+, F–, L+, and L– productions. Each pair of deductions combines a context of one or more connected component states with a sign (word) observed in that context. By applying the F and L rules to the observed sign and context, the parser is able to generate a consequent context. Initially, the context corresponds to a connected pre-sentential dialogue state $\top/\top$. When *the* is observed, the parser applies F+ to begin a new connected component state D. By applying L–, the parser determines that this new connected component is unfinished and generates an appropriate incomplete connected component state NP/N, encoding the superordinate state $\top/\top$ for later retrieval. Further on, the parser observes *'s* and uses F– to avoid generating a new connected component, which completes the sign D. The parser follows this up with L+ to recall the superordinate connected component state S/NP and integrate it into the most deeply embedded connected component, which results in a less deeply embedded structure.

child of the awaited sign of the immediately superordinate connected component state a/b, it is hypothesized to do so and terminate the subordinate connected component state, with x_t as the last observation of the terminated connected component (L+); or if the observation can serve as a lower descendant of this awaited sign, it is hypothesized to remain disjoint and form its own connected component (L–):

$$\frac{a/b \quad a''}{a/b''}\; b \rightarrow a''\; b'' \qquad \text{(L+)}$$

$$\frac{a/b \quad a''}{a/b \quad a'/b''}\; b \xrightarrow{+} a' \ldots ; \;\; a' \rightarrow a''\; b'' \qquad \text{(L–)}$$

These operations can be made probabilistic. The probability σ of a transition at time step t is defined in terms of (i) a probability ϕ of initiating a new connected component state with x_t as its first observation, multiplied by (ii) the probability λ of terminating a connected component state with x_t as its last observation, multiplied by (iii) the probabilities α and β of generating categories for active and awaited signs $a_{q_t^n}$ and $b_{q_t^n}$ in the resulting most subordinate connected component state q_t^n. This kind of model can be defined directly on PCFG probabilities and trained to produce state-of-the-art accuracy by using the latent variable annotation of Petrov et al. (2006) (van Schijndel et al., in press).[5]

An example parse is shown in Figure 2. Since two binary structural decisions (F and L) must be made in order to generate each word, there are four possible structures that may be generated (see Table 1). The F+L– transition initiates a new level of embedding at word x_t and so requires the superordinate state to be encoded for later retrieval (e.g. on observing *the* in Figure 2). The F–L+ transition completes the deepest level of embedding and therefore requires the recall of the current superordinate connected component state with which the

[5]The model has been shown to achieve an F-score of 87.8, within .2 points of the Petrov and Klein (2007) parser, which obtains an F-score of 88.0 on the same task. Because the sequence model is defined over binary-branching phrase structure, both parsers were evaluated on binary-branching phrase structure trees to provide a fair comparison.

F–L–	Cue Active Sign
F+L–	Initiate/Encode
F–L+	Terminate/Integrate
F+L+	Cue Awaited Sign

Table 1: The hierarchical structure decisions and the operations they represent. F+L– initiates a new connected component, F–L+ integrates two disjoint connected components into a single connected component, and F–L– and F+L+ sequentially cue, respectively, a new active sign (along with an associated awaited sign) and a new awaited sign from the most recent connected component.

Theory	F+L–	F–L+
DLT	positive	positive
ACT-R	positive	positive
Hier. Sequential Prediction		positive
Dynamic Recruitment	positive	negative
Embedding Difference	positive	negative

Table 2: Each theory's prediction of the direction of the correlation between each hierachical structure predictor and reading times. Hierarchic sequential prediction is agnostic about the processing speed of F+L– operations, and none of the theories make any predictions as to the sign associated with the within-embedding measures F–L– and F+L+.

subordinate connected component state will be integrated. For example, in Figure 2, upon observing *'s*, the parser must use temporal cueing to recall that it is in the middle of processing an NP (to complete an S), which sequentially cues a prediction of N. F–L– transitions complete the awaited sign of the most subordinate state and so sequentially cue a following connected component state at the same tier of the hierarchy. For example, in Figure 2, after observing *studio*, the parser uses the completed NP to sequentially cue the prediction that it has finished the left child of an S. F+L+ transitions locally expand the awaited sign of the most subordinate state and so should also not require any recall or encoding. For example, in Figure 2, observing *bought* while awaiting a VP sequentially cues a prediction of NP.

F+L–, then, loosely corresponds to a *storage* action under DLT as more hierarchic levels must now be maintained at each future step of the parse. As stated before, it differs from DLT in that it is sensitive to the depth of embedding rather than a particular subset of syntactic categories. Wu et al. (2010) found that increasing the embedding depth led to longer reading times in a self-paced reading experiment. In ACT-R terms, F+L– corresponds to an *encoding* action, potentially causing processing difficulty resulting from the similarity of the current sign to previously encoded signs.

F–L+, by contrast, is similar to DLT's *integration* action since a subordinate connected component is integrated into the rest of the parse structure. This represents a temporal cueing event in which the awaited category of the superordinate connected component is recalled. In contrast to DLT, embedding difference and dynamic recruitment would predict a *shorter* reading time in the F–L+ case because of the reduction in memory load. In an ACT-R framework, reading time durations can increase at the retrieval site because the retrieval causes competition among similarly encoded signs in the context set. While it is possible for reading times to decrease when completing a center embedding in ACT-R (Vasishth and Lewis, 2006), this would be expressed as a frequency effect due to certain argument types commonly foreshadowing their predicates (Jaeger et al., 2008). Since frequency effects are factored separately from memory effects in this study, ACT-R would predict longer residual (memory-based) reading times when completing an embedding.

Predicted correlations to reading times for the F and L transitions are summarized in Table 2.

5 Eye-tracking

Eye-tracking and reading time data are often used to test complexity measures (Gibson, 2000; Demberg and Keller, 2008; Roark et al., 2009) under the assumption that readers slow down when reading more complex passages. Readers saccade over portions of text and regress back to preceding text in complex patterns, but studies have correlated certain measures with certain processing constraints (see Clifton et al. 2007 for a review). For example, the initial length of time fixated on a single word is correlated with word identification time; whereas regression durations after a word is fixated (but prior to a fixation in a new region) are hypothesized to correlate

with integration difficulty.

Since this work focuses on incremental processing, all processing that occurs up to a given point in the sentence is of interest. Therefore, in this study, predictions will be compared to *go-past durations*. Go-past durations are calculated by summing all fixations in a region of text, including regressions, until a new region is fixated, which accounts for additional processing that may take place after initial lexical access, but before the next region is processed. For example, if one region ends at word 5 in a sentence, and the next fixation lands on word 8, then the *go-past region* consists of words 6-8 and the *go-past duration* sums all fixations until a fixation occurs after word 8.

6 Evaluation

The measures presented in this paper were evaluated on the Dundee eye-tracking corpus (Kennedy et al., 2003). The corpus consists of 2388 sentences of naturally occurring news text written in standard British English. The corpus also includes eye-tracking data from 10 native English speakers, which provides a test corpus of 260,124 subject-duration pairs of reading time data. Of this, any fixated words appearing fewer than 5 times in the training data were considered unknown and were filtered out to obtain accurate predictions. Fixations on the first or last words of a line were also filtered out to avoid any 'wrap-up' effects resulting from preparing to saccade to the beginning of the next line or resulting from orienting to a new line. Additionally, following Demberg and Keller (2008), any fixations that skip more than 4 words were attributed to track loss by the eyetracker or lack of attention of the reader and so were excluded from the analysis. This left the final evaluation corpus with 151,331 subject-duration pairs.

The evaluation consisted of fitting a linear mixed-effects model (Baayen et al., 2008) to reading time durations using the *lmer* function of the *lme4* R package (Bates et al., 2011; R Development Core Team, 2010). This allowed by-subject and by-item variation to be included in the initial regression as random intercepts in addition to several baseline predictors.[6] Before fitting, the durations extracted from the corpus were log-transformed, producing more normally distributed data to obey the assumptions of linear mixed effects models.[7]

Included among the fixed effects were the position in the sentence that initiated the go-past region (SENTPOS) and the number of characters in the initiating word (NRCHAR). The difficulty of integrating a word may be seen in whether the immediately following word was fixated (NEXTISFIX), and similarly if the immediately previous word was fixated (PREVISFIX) the current word probably need not be fixated for as long. Finally, unigram (LOGPROB) and bigram probabilities are included. The bigram probabilities are those of the current word given the previous word (LOGFWPROB) and the current word given the following word (LOGBWPROB). Fossum and Levy (2012) showed that for n-gram probabilities to be effective predictors on the Dundee corpus, they must be calculated from a wide variety of texts, so following them, this study used the Brown corpus (Francis and Kucera, 1979), the WSJ Sections 02-21 (Marcus et al., 1993), the written text portion of the British National Corpus (BNC Consortium, 2007), and the Dundee corpus (Kennedy et al., 2003). This amounted to an n-gram training corpus of roughly 87 million words. These statistics were smoothed using the SRILM (Stolcke, 2002) implementation of modified Kneser-Ney smoothing (Chen and Goodman, 1998). Finally, total surprisal (SURP) was included to account for frequency effects in the baseline.

The preceding measures are commonly used in baseline models to fit reading time data (Demberg and Keller, 2008; Frank and Bod, 2011; Fossum and Levy, 2012) and were calculated from the final word of each go-past region. The following measures create a more sophisticated baseline by accumulating over the entire go-past region to capture what must be integrated into the discourse to continue the parse. One factor (CWDELTA) simply counts the number of words in each go-past region. Cumula-

[6] Each fixed effect was centered to reduce collinearity.

[7] In particular, these models assume the noise in the data is normally distributed. Initial exploratory trials showed that the residuals of fitting any sensible baseline also become more normally distributed if the response variable is log-transformed. Finally, the directions of the effects remain the same whether or not the reading times are log-transformed, though significance cannot be ascertained without the transform.

tive total surprisal (CUMUSURP) and cumulative entropy reduction (ENTRED) give the surprisal (Hale, 2001) and entropy reduction (Hale, 2003) summed over the go-past region. To avoid convergence issues, each of the cumulative measures is residualized from the next simpler model in the following order: CWDELTA from the standard baseline, CUMUSURP from the baseline with CWDELTA, and ENTRED from the baseline with all other effects.

Residualization was accomplished by using the simpler mixed-effects model to fit the measure of interest. The residuals from that model fit were then used in place of the factor of interest. All joint interactions were included in the baseline model as well. Finally, to account for spillover effects (Just et al., 1982) where processing from a previous region contributes to the following duration, the above baseline predictors from the previous go-past region were included as factors for the current region.

Having SURP as a predictor with CUMUSURP may seem redundant, but initial analyses showed SURP was a significant predictor over CUMUSURP when CWDELTA was a separate factor in the baseline (current: $p = 2.2 \cdot 10^{-16}$ spillover: $p = 2 \cdot 10^{-15}$) and vice versa (current: $p = 2.2 \cdot 10^{-16}$ spillover: $p = 6 \cdot 10^{-5}$). One reason for this could be that go-past durations conflate complexity experienced when initially fixating on a region with the difficulty experienced during regressions. By including both versions of surprisal, the model is able to account for frequency effects occurring in both conditions.

This study is only interested in how well the proposed memory-based measures fit the data over the baseline, so to avoid fitting to the test data or weakening the baseline by overfitting to training data, the full baseline was used in the final evaluation.

Each measure proposed in this paper was summed over go-past regions to make it cumulative and was residualized from all non-spillover factors before being included on top of the full baseline as a main effect. Likewise, the spillover version of each proposed measure was residualized from the other spillover factors before being included as a main effect. Only a single proposed measure (or its spillover corrollary) was included in each model. The results shown in Table 3 reflect the probability of the full model fit being obtained by the model lacking each factor of interest. This was found via posterior sampling of each factor using the Markov chain Monte Carlo implementation of the *languageR* R package (Baayen, 2008).

Factor	Operation	t-score	p-value
F–L–	Cue Active	0.60	0.55
F+L–	Initiate	7.10	$2.22 \cdot 10^{-14}$
F–L+	Integrate	-5.44	$5.23 \cdot 10^{-8}$
F+L+	Cue Awaited	-1.55	0.12

Table 3: Significance of each of the structure generation outcomes at predicting log-transformed durations when added to the baseline as a main effect after being residualized from it. The sign of the t-score indicates the direction of the correlation between the residualized factor and go-past durations. Note that these factors are all based on the current go-past region; the spillover corollaries of these were not significant predictors of reading times.

The results indicate that the F+L– and F–L+ measures were both significant predictors of duration as expected. Further, F–L– and F+L+, which both simply reflect sequential cueing, were not significant predictors of go-past duration, also as expected.

7 Discussion and Conclusion

The fact that F+L– was strongly predictive over the baseline is encouraging as it suggests that memory limitations could provide at least a partial explanation of *why* certain constructions are less frequent in corpora and thus yield a high surprisal. Moreover, it indicates that the model corroborates the shared prediction of most of the memory-based models that initiating a new connected component slows processing.

The fact that F–L+ is predictive but has a negative coefficient could be evidence of anti-locality, or it could be an indication of some sort of processing momentum due to dynamic recruitment of additional processing resources (Just and Varma, 2007). Since anti-locality is an expectation-based frequency effect, and since this study controlled for frequency effects with n-grams, surprisal, and entropy reduction, an anti-locality explanation would rely on either (i) more precise variants of the metrics used in this study or (ii) other frequency metrics altogether. Future work could investigate the possibility of anti-locality by looking at the distance between an encoding operation and its corresponding

integration action to see if the integration facilitation observed in this study is driven by longer embeddings or if there is simply a general facilitation effect when completing embeddings.

The finding of a negative integration cost was previously observed by Wu et al. (2010) as well as Demberg and Keller (2008), although Demberg and Keller calculated it using the original referent-based definitions of Gibson (1998; 2000) and varied which parts of speech counted for calculating integration cost. Ultimately, Demberg and Keller (2008) concluded that the negative coefficient was evidence that integration cost was not a good broad-coverage predictor of reading times; however, this study has replicated the effect and showed it to be a very strong predictor of reading times, albeit one that is correlated with facilitation rather than inhibition.

It is interesting that many studies have found negative integration cost using naturalistic stimuli while others have consistently found positive integration cost when using constructed stimuli with multiple center embeddings presented without context (Gibson, 2000; Chen et al., 2005; Kwon et al., 2010). It may be the case that any dynamic recruitment is overwhelmed by the memory demands of multiply center-embedded stimuli. Alternatively, it may be that the difficulty of processing multiply center-embedded sentences containing ambiguities produces anxiety in subjects, which slows processing at implicit prosodic boundaries (Fodor, 2002; Mitchell et al., 2008). In any case, the source of this discrepancy presents an attractive target for future research.

In general, sequential prediction does not seem to present people with any special ease or difficulty as evidenced by the lack of significance of F–L– and F+L+ predictions when frequency effects are factored out. This supports a theory of sequential, content-based cueing (Botvinick, 2007) that predicts that certain states would directly cue other states and thus avoid recall difficulty. An example of this may be seen in the case of a transitive verb triggering the prediction of a direct object. This kind of cueing would show up as a frequency effect predicted by surprisal rather than as a memory-based cost, due to frequent occurrences becoming ingrained as a learned skill. Future work could use these sequential cueing operations to investigate further claims of the dynamic recruitment hypothesis. One of the implications of the hypothesis is that recruitment of resources alleviates the initial encoding cost, which allows the parser to continue on as before the embedding. DLT, on the other hand, predicts that there is a storage cost for maintaining unresolved dependencies during a parse (Gibson, 2000). By weighting each of the sequential cueing operations with the embedding depth at which it occurs, an experiment may be able to test these two predictions.

This study has shown that measures based on working memory operations have strong predictivity over other previously proposed measures including those associated with frequency effects. This suggests that memory limitations may provide a partial explanation of what gives rise to frequency effects. Lastly, this paper provides evidence that there is a robust facilitation effect in English that arises from completing center embeddings.

The hierarchic sequence model, all evaluation scripts, and regression results for all baseline predictors used in this paper are freely available at http://sourceforge.net/projects/modelblocks/.

Acknowledgements

Thanks to Peter Culicover, Micha Elsner, and three anonymous reviewers for helpful suggestions. This work was funded by an OSU Department of Linguistics Targeted Investment for Excellence (TIE) grant for collaborative interdisciplinary projects conducted during the academic year 2012-13.

References

John R. Anderson, Dan Bothell, Michael D. Byrne, S. Douglass, Christian Lebiere, and Y. Qin. 2004. An integrated theory of the mind. *Psychological Review*, 111(4):1036–1060.

R. Harald Baayen, D. J. Davidson, and Douglas M. Bates. 2008. Mixed-effects modeling with crossed random effects for subjects and items. *Journal of Memory and Language*, 59:390–412.

R. Harald Baayen. 2008. *Analyzing Linguistic Data: A Practical Introduction to Statistics using R*. Cambridge University Press, New York, NY.

Douglas Bates, Martin Maechler, and Ben Bolker, 2011. *lme4: Linear mixed-effects models using S4 classes*.

BNC Consortium. 2007. The british national corpus.

Matthew Botvinick. 2007. Multilevel structure in behavior and in the brain: a computational model of Fuster's hierarchy. *Philosophical Transactions of the Royal Society, Series B: Biological Sciences*, 362:1615–1626.

Stanley F. Chen and Joshua Goodman. 1998. An empirical study of smoothing techniques for language modeling. Technical report, Harvard University.

Evan Chen, Edward Gibson, and Florian Wolf. 2005. Online syntactic storage costs in sentence comprehension. *Journal of Memory and Language*, 52(1):144–169.

Charles Clifton, Adrian Staub, and Keith Rayner. 2007. Eye movements in reading words and sentences. In *Eye movements: A window on mind and brain*, pages 341–372. Elsevier.

Vera Demberg and Frank Keller. 2008. Data from eye-tracking corpora as evidence for theories of syntactic processing complexity. *Cognition*, 109(2):193–210.

Janet Fodor. 2002. Prosodic disambiguation in silent reading. In M. Hirotani, editor, *In Proceedings of NELS 32*.

Victoria Fossum and Roger Levy. 2012. Sequential vs. hierarchical syntactic models of human incremental sentence processing. In *Proceedings of CMCL-NAACL 2012*. Association for Computational Linguistics.

W. Nelson Francis and Henry Kucera. 1979. The brown corpus: A standard corpus of present-day edited american english.

Stefan Frank and Rens Bod. 2011. Insensitivity of the human sentence-processing system to hierarchical structure. *Psychological Science*.

Edward Gibson. 1998. Linguistic complexity: Locality of syntactic dependencies. *Cognition*, 68(1):1–76.

Edward Gibson. 2000. The dependency locality theory: A distance-based theory of linguistic complexity. In *Image, language, brain: Papers from the first mind articulation project symposium*, pages 95–126.

John Hale. 2001. A probabilistic earley parser as a psycholinguistic model. In *Proceedings of the second meeting of the North American chapter of the Association for Computational Linguistics*, pages 159–166, Pittsburgh, PA.

John Hale. 2003. *Grammar, Uncertainty and Sentence Processing*. Ph.D. thesis, Cognitive Science, The Johns Hopkins University.

John Hale. 2006. Uncertainty about the rest of the sentence. *Cognitive Science*, 30(4):609–642.

John Hale. 2011. What a rational parser would do. *Cognitive Science*, 35(3):399–443.

Marc W. Howard and Michael J. Kahana. 2002. A distributed representation of temporal context. *Journal of Mathematical Psychology*, 45:269–299.

F. T. Jaeger, E. Fedorenko, P. Hofmeister, and E. Gibson. 2008. Expectation-based syntactic processing: Antilocality outside of head-final languages. In *The 21st CUNY Sentence Processing Conference*.

Daniel Jurafsky. 1996. A probabilistic model of lexical and syntactic access and disambiguation. *Cognitive Science: A Multidisciplinary Journal*, 20(2):137–194.

Marcel Adam Just and Sashank Varma. 2007. The organization of thinking: What functional brain imaging reveals about the neuroarchitecture of complex cognition. *Cognitive, Affective, & Behavioral Neuroscience*, 7:153–191.

Marcel Adam Just, Patricia A. Carpenter, and Jacqueline D. Woolley. 1982. Paradigms and processes in reading comprehension. *Journal of Experimental Psychology: General*, 111:228–238.

Alan Kennedy, James Pynte, and Robin Hill. 2003. The Dundee corpus. In *Proceedings of the 12th European conference on eye movement*.

Nayoung Kwon, Yoonhyoung Lee, Peter C. Gordon, Robert Kluender, and Maria Polinsky. 2010. Cognitive and linguistic factors affecting subject/object asymmetry: An eye-tracking study of pre-nominal relative clauses in korean. *Language*, 86(3):561.

Roger Levy. 2008. Expectation-based syntactic comprehension. *Cognition*, 106(3):1126–1177.

Richard L. Lewis and Shravan Vasishth. 2005. An activation-based model of sentence processing as skilled memory retrieval. *Cognitive Science*, 29(3):375–419.

Richard L. Lewis, Shravan Vasishth, and Jane A. Van Dyke. 2006. Computational principles of working memory in sentence comprehension. *Trends in Cognitive Science*, 10(10):447–454.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Brian McElree. 2001. Working memory and focal attention. *Journal of Experimental Psychology, Learning Memory and Cognition*, 27(3):817–835.

Brian McElree. 2006. Accessing recent events. *The Psychology of Learning and Motivation*, 46:155–200.

D. Mitchell, X. Shen, M. Green, and T. Hodgson. 2008. Accounting for regressive eye-movements in models of sentence processing: A reappraisal of the selective reanalysis hypothesis. *Journal of Memory and Language*, 59:266–293.

Slav Petrov and Dan Klein. 2007. Improved inference for unlexicalized parsing. In *Proceedings of NAACL HLT 2007*, pages 404–411, Rochester, New York, April. Association for Computational Linguistics.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 44th Annual Meeting of the Association for Computational Linguistics (COLING/ACL'06)*.

Colin Phillips. 2010. Some arguments and non-arguments for reductionist accounts of syntactic phenomena. *Language and Cognitive Processes*, 28:156–187.

R Development Core Team, 2010. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.

Brian Roark, Asaf Bachrach, Carlos Cardenas, and Christophe Pallier. 2009. Deriving lexical and syntactic expectation-based measures for psycholinguistic modeling via incremental top-down parsing. *Proceedings of the 2009 Conference on Empirical Methods in Natural Langauge Processing*, pages 324–333.

Brian Roark. 2001. Probabilistic top-down parsing and language modeling. *Computational Linguistics*, 27(2):249–276.

Claude Shannon. 1948. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656.

Stuart M. Shieber, Yves Schabes, and Fernando C.N. Pereira. 1995. Principles and implementation of deductive parsing. *Journal of Logic Programming*, 24:3–36.

Andreas Stolcke. 2002. Srilm – an extensible language modeling toolkit. In *Seventh International Conference on Spoken Language Processing*.

John Trueswell. 1996. The role of lexical frequency in syntactic ambiguity resolution. *Journal of Memory and Language*, 35:566–585.

Marten van Schijndel, Andy Exley, and William Schuler. in press. A model of language processing as hierarchic sequential prediction. *Topics in Cognitive Science*.

Shravan Vasishth and Richard L. Lewis. 2006. Argument-head distance and processing complexity: Explaining both locality and antilocality effects. *Language*, 82(4):767–794.

Stephen Wu, Asaf Bachrach, Carlos Cardenas, and William Schuler. 2010. Complexity metrics in an incremental right-corner parser. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL'10)*, pages 1189–1198.