An eye-tracking evaluation of some parser complexity metrics

Matthew J. Green
University of Aberdeen, UK
mjgreen@abdn.ac.uk

Abstract

Information theoretic measures of incremental parser load were generated from a phrase structure parser and a dependency parser and then compared with incremental eye movement metrics collected for the same temporarily syntactically ambiguous sentences, focussing on the disambiguating word. The findings show that the *surprisal* and *entropy reduction* metrics computed over a phrase structure grammar make good candidates for predictors of text readability for human comprehenders. This leads to a suggestion for the use of such metrics in Natural Language Generation (NLG).

1 Introduction

This work aims to predict automatically how difficult a generated sentence will be for a person to read. Temporarily syntactically ambiguous sentences were presented along with pre-disambiguated controls to people to read while their eye movements were recorded. The same materials were given as input to two NLP parsers, trained on portions of the Wall Street Journal part of the Penn Treebank, that generate incremental word by word metrics of parser load. The metrics of parser load were compared against a standard measure of human sentence processing load *regression path duration*.

The purpose of the present article is to demonstrate that the parser metrics can predict human difficulty for a certain syntactically-ambiguous sentence type (described in the next section). The article also proposes that, if future work shows that the parser metrics here also predict sentence processing difficulty more broadly, then this method would be a useful way for NLG systems to decide on a particular output from among several possible outputs that express the same information.

2 Complement ambiguity

The sentences used in this article were representative of *complement ambiguity*. Sentences like these are syntactically ambiguous until a disambiguating word, which resolves the ambiguity either to *no complement*, *direct object complement*, or *sentential complement*. This section gives the linguistic aspects of this ambiguity type with examples. Material in parentheses indicates how the unambiguous controls were constructed: by means of punctuation indicating the clause boundary in (1); and by means of an overt complementiser establishing the sentential complement in (2). Phrase marker diagrams are given for the examples in Figures (1) and (2).

(1) After the cadet saluted(,) the captain walked to the gates of the enclosure. SENTENCE TYPE 1

(2) The cadet noticed (that) the captain walked to the gates of the enclosure. SENTENCE TYPE 2

Sentential complement ambiguities exploit the properties of 'complement' verbs like *noticed* that can be followed either by a complement clause or by a direct object, or by no complement. When such verbs are followed by complements and an overt complementiser like *that* is used, no temporary syntactic ambiguity is present: however, when the complementiser is omitted, which may be done without violating the grammar, temporary syntactic ambiguity arises with respect to the first few words of the complement. These words may be taken as a direct object instead, and then when the complement verb appears, disambiguation ensues as the words that were taken to be part of a direct object of the verb are revealed necessarily to be part of a complement. Another possibility afforded by the multiple subcategorisation frame

Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR) @ EACL 2014, pages 38–46,
Gothenburg, Sweden, April 26-30 2014.

of words like *noticed* is that the words immediately following could properly be the start of a main clause where the clause containing *noticed* is properly a subordinate clause. Such cases are sometimes referred to as reduced complements. In these cases only the presence of a main verb resolves the temporary syntactic ambiguity, and when it appears, some major restructuring is involved. Complement ambiguities of both kinds have been used to investigate the parsing of ambiguous clauses (Holmes et al., 1987; Rayner and Frazier, 1987; Sturt et al., 1999; Ferreira and Henderson, 1991; Clifton Jr, 1993; Pickering and Traxler, 1998; Trueswell et al., 1993).

Evidence from studies with human readers support the notion that there is a processing difficulty differential across the two forms such that disambiguation in sentence type (1) is harder than in sentence type (2). This has been shown using grammaticality judgements (Ferreira and Henderson, 1991), self-paced reading times (Sturt et al., 1999), and eye-tracking (Green, 2014).

The current article presents an eye-tracking evaluation of the parser predictions for complement ambiguity, and discusses applications of syntactic complexity metrics for evaluating test readability.

3 Parser metrics

This section gives details of the *surprisal*, *entropy reduction*, and *retrieval time* metrics, and how they are computed.

3.1 Surprisal

Surprisal was computed over a phase structure parser, and over a dependency parser.

Surprisal is computed using two other quantities. These quantities are: (1) the probability of a derivation: a derivation is a set of weighted rule productions that result in the current partial string of input words, such that a sentence fragment with two alternative parses is represented as two derivations; (2) prefix probability: this is the probability of the parse of the fragment seen so far, which is composed of the sum of the probabilities of the two derivations if the fragment is syntactically ambiguous with two alternatives.

Let G be a probabilistic context free grammar (PCFG). Let d be a derivation composed of a sequence of applications of grammar rules. Let i index these applications so that d_i is the ith application in d, and let j be the total number of applications in the derivation. Then the probability p of a derivation d given a grammar G and the current sentence fragment $w_{1\ldots k}$ is given by the product of the probability of each rule applied in the derivation, thus:

$$\mathrm{p}(d, G, w_{1\ldots k}) = \prod_{i=1}^{j} \mathrm{p}(d_i, G, w_{1\ldots k})$$

Let $\mathcal{D}$ represent the set of all derivations d that are present for the current sentence fragment – when there are two alternative parses available for the sentence fragment seen so far, $\mathcal{D}$ has two elements. Let w be the set of words in the sentence fragment seen so far. Let w_k be the word that the parser encountered most recently at the current state. Let w_{k+1} be the first word of the rest of the sentence. As the parser transitions from its state at w_k to its state at w_{k+1} we can derive a *prefix probability* pp at w_{k+1} that represents the sum probability of the derivations of the string $w_{1\ldots k+1}$. So the prefix probability of word w_{k+1} with respect to a probabilistic context free grammar (PCFG) denoted G is given by the sum of the probability of all derivations of the string $w_{1\ldots k+1}$ that the grammar generates.

$$\mathrm{pp}(w_{k+1}, G, w_{1\ldots k}) = \sum_{d \in \mathcal{D}} \mathrm{p}(d, G, w_{1\ldots k})$$

The conditional probability cp of the next word w_{k+1} is the ratio of the prefix probability of the next word w_{k+1} to the prefix probability of the current word w_k.

$$\mathrm{cp}(w_{k+1}, G, w_{1\ldots k}) = \frac{\mathrm{pp}(w_{k+1}, G, w_{1\ldots k})}{\mathrm{pp}(w_k, G, w_{1\ldots k-1})}$$

The surprisal sp, measured in *bits* of information, associated with the next word w_{k+1} is the negative log of the conditional probability of the next word w_{k+1}

$$\mathrm{sp}(w_{k+1}, G, w_{1\ldots k}) = -\log(\mathrm{cp}(w_{k+1}, G, w_{1\ldots k}))$$

The TDPARSE top-down incremental parser provided by Roark (2013) and described in Roark (2001) and Roark (2004) computes surprisal over a phrase structural grammar, incrementally for each word in a sentence. It is a parallel parser that maintains potentially very many parses at each state. For details of how the beam width varies across a sentence, see Roark (2001).

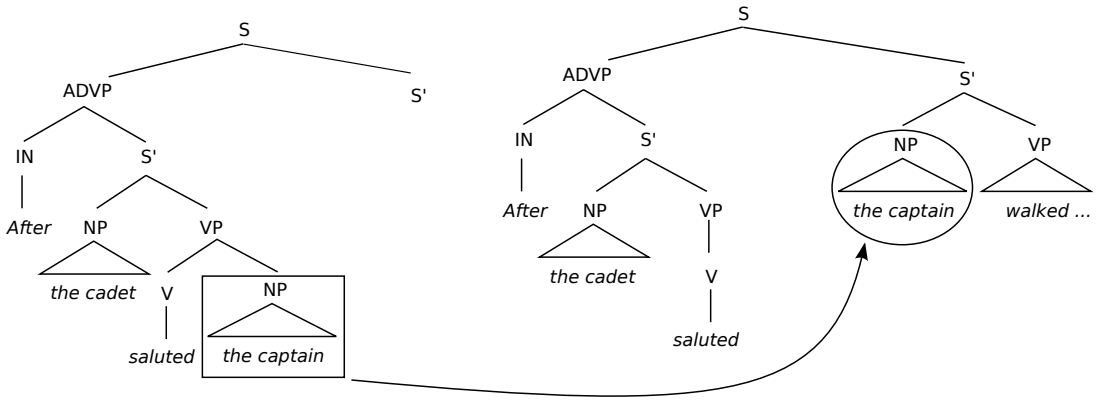

Figure 1: Phrase markers showing disambiguation in sentence type 1. The left phrasemarker shows the initial misattachment. The right phrasemarker shows how the same initially misattached NP is attached in the ultimately correct analysis.

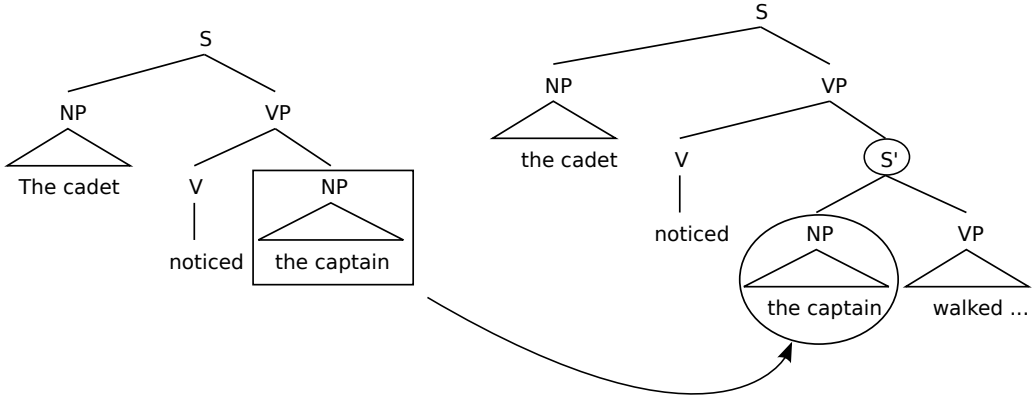

Figure 2: Phrase markers showing disambiguation in sentence type 2. The left phrasemarker shows the initial misattachment. The right phrasemarker shows how the same initially misattached NP is attached in the ultimately correct analysis.

The HUMDEP parser provided by Boston (2013) and described in Boston and Hale (2007) and Boston (2012) computes surprisal over a dependency grammar transition system , incrementally for each word in a sentence. It is a k-best parser. Here the value of k was set to 3, in line with previous use of the parser to model human disambiguation performance for garden-path sentences in Boston and Hale (2007).

Hypothesis 1 Hale (2001), and also Levy (2008), gave the hypothesis that surprisal is linearly related to the human effort of processing a particular word in a sentence fragment. This hypothesis casts disambiguation as the work incurred by disconfirming all parses of the fragment that are inconsistent with the fragment including the disambiguating word.

3.2 Entropy reduction

Entropy reduction was computed over the output of the phrase structure parser TDPARSE. In general, the entropy (Shannon, 1948), denoted H, of a random variable is the uncertainty associated with that variable. Specifically, for a discrete random variable X with outcomes $x_1, x_2, \ldots$ with probabilities $p_1, p_2, \ldots$

$$\mathrm{H}(X) = -\sum_{x \in X} p_x \log_2 p_x$$

Putting this in sentence processing terms, let D be a set of derivations for a sentence fragment W and let X be the extended sentence fragment that results from adding a new word to the fragment.

$$H(\mathcal{G}, D, W) = -\sum prp(\mathcal{G}, X) log(prp(\mathcal{G}, X))$$

The quantity *entropy reduction* is defined with a lower bound of zero so that this quantity is never

negative:

$$\mathrm{ER} = max(0, \mathrm{H}(\mathcal{D}|w_{1...k}) - \mathrm{H}(\mathcal{D}|w_{1...k+1}))$$

Hypothesis 2 Hale (2004) and Hale (2006) gave the *entropy reduction hypothesis* that the human effort of processing a particular word in a sentence fragment is the reduction in entropy from its value given the fragment to its value given the fragment including the disambiguating word.

3.3 Retrieval time

Parsing in retrieval theory (Lewis and Vasishth, 2005) is accomplished by condition-action pairs generated with reference to a phrase structure grammar. A series of memory buffers stores elements in short-term and long-term buffers. Parallel associative retrieval (McElree et al., 2003), fluctuation of activation of elements already in a memory buffer, and retrieval interference as a function of similarity are combined to predict the amount of time that it takes to read a word (Vasishth et al., 2008).

A word's activation is based on two quantities: the baseline activation of the word, which is taken to decay given the passage of time; and the amount of similarity based interference with other words that have been parsed. The baseline activation B for a word i is given here, taken from Lewis and Vasishth (2005), and Patil et al. (2009), where t_r is the time since the rth retrieval of the word, the summation is over all n retrievals, and d is a decay factor set to 0.5 as in other ACT-R models (Anderson, 2005).

$$B_i = \ln\left(\sum_{r=1}^{n} t_r - d\right)$$

The equation tracks the log odds that a word will need to be retrieved, given its past usage history. It yields not a smoothly decaying activation from initial encoding to the current time, but a "series of spikes corresponding to the retrieval events" (Lewis and Vasishth, 2005).

The overall activation A for word i is given here

$$A_i = B_i + \sum_j W_j S_{ji}$$

from Lewis and Vasishth (2005). In this equation, B_i is the fluctuating baseline level of activation for word i which is subject to time-based decay. In the model, a *goal buffer* contains retrieval cues for integrating the current word. Overall activation A for word i is found by adding to the baseline activation for word i an associative activation boost received from retrieval cues in the goal buffer that are associated with i. The variable j indexes those retrieval cues in the goal buffer. W_js are weights on the retrieval cues in the goal buffer. The weight on a retrieval cue represents the proportion of the total activation available for the whole goal buffer that is assigned to the particular retrieval cue j in the goal buffer. S_{ji}s are the strengths of association from each retrieval cue j of the goal buffer to word i. This equation is effectively adding to the baseline activation an activation boost received from retrieval cues in the goal buffer.

The amount of similarity based interference is estimated by the weighted strengths of association between the word to be retrieved and retrieval cues from other words already parsed and with a trace in memory. In the following equation, word i is the current word, and retrieval cue j is from a word that is similar to word i, with reference to its part of speech tag, so that nouns interfere with other nouns but not with verbs. If retrieval cue j is similar to word i then the amount by which retrieval cue j interferes with word i varies according to how many words have already been associated with retrieval cue j. The array of words that is associated with retrieval cue j is considered to form a fan so that fan_j gives the number of words in the fan for cue j. The constant S refers to the maximum associative strength of 1.5 (Lewis and Vasishth, 2005).

$$S_{ji} = S - \ln(fan_j)$$

This equation is effectively reducing the maximum associative strength S by the log of the "fan" of cue j, that is, the number of items associated with j.

The mapping from activation level to retrieval time is given next. F is a scaling constant set to 0.14 in Lewis and Vasishth (2005). A_i is the word's activation and e is Euler's constant. T_i is retrieval time for word i:

$$T_i = Fe^{A_i}$$

The retrieval time measure comes from Lewis and Vasishth (2005) where a theory of sentence processing is expressed as set of processes corresponding with skilled retrievals of linguistic components from memory. However in that paper it is computed over a phrase structure gram-

mar. Boston provides a method to compute retrieval time over a dependency grammar in the HUMDEP3.0 parser and Boston's method (Boston, 2013) is used here.

Hypothesis 3 Retrieval time is related to human sentence processing difficulty.

4 Eye movement metrics

This section gives the metrics used to index human sentence processing load at disambiguation. Rayner et al. (2012, p. 93) set out the most common eye tracking measures. These include the following measures: First Fixation Duration (FFD); First Pass Reading Time (FPRT); Regression Path Duration (RPD). These are defined next. First fixation duration (FFD) is the mean duration of the first fixation on a word regardless of other possible fixations on the word. It has traditionally been treated as a measure of early processing. First fixation duration is interpreted to index lexical access. First pass reading time (FPRT): also known as gaze duration, is the sum of the durations of all fixations on the word that occur before leaving the word in any direction. This still captures the early processing (FFD is a subset of FPRT) but FPRT also includes any refixations that there might be on the word before a regression is launched from it. First pass reading time is often interpreted to index lexical integration into the phrase marker. Regression path duration (RPD) includes FPRT but adds to it the durations of fixations on preceding words that the eyes regress to before leaving the word to the right to take in new material, as well as any refixations on the launch word that occur before new material is taken in. In this way RPD is sensitive to integration difficulties that yield regressive eye movements but it also includes early processing. Regression path duration is often interpreted to index incremental syntactic integration of the new word into the sentence's representation including any semantic problems that arise from this.

Since RPD is the measure most sensitive to syntactic disambiguation, it is used in this article as a measure that is representative of human parsing load at disambiguation.

5 Method

This section tells how the eye tracking experiment was carried out.

Participants were forty native speakers of British English who were students of Psychology at the University of Exeter and who participated for course credit. All had normal or corrected-t-normal vision, were naive as to the purpose of the experiment, aged between eighteen and thirty-four.

Apparatus used was an SR Research EyeLink II head-mounted eyetracker. This recorded participants' eye movements with a sampling rate of 500 Hz while they read sentences displayed on a 19 inch Iiyama Vision Master Pro monitor at 1024 x 768 resolution at a refresh rate of 60 Hz. Viewing was binocular but only the right eye was recorded. Participants sat in a dimly lit room in front of the computer at a viewing distance of approximately 75 cm the average viewing distance was approximately 75 cm. At this viewing distance, and assuming that 1 character had 2 mm width on screen, a single character subtended 0.153 degrees of visual angle, and approximately 6.5 characters subtended 1 degree of visual angle. The font used was Courier New 12 point. All sentences in this experiment were displayed on a single line with a maximum length of 100 characters. A 9 point calibration procedure was used, on which participants were required to achieve a score of 'good'. Each trial started with a drift correction routine where the participant was required to fixate a target that appeared in the same location as the first character of the sentence would subsequently occupy, and then required to press a button on the gamepad while fixating this point to start the trial.

Participants were instructed to read silently for comprehension at a comfortable speed. The practice trials and experimental trials were implemented as separate consecutive blocks. The experimental trials were randomised by Experiment Builder each time the experiment was run, i.e., in a different order for each participant, with the constraint that a maximum of two trials of a given type could appear in a continuous sequence. There were four practice sentences, followed by a drift correction routine preceding the experimental block containing 96 sentences, comprising 24 in experimental conditions (6 in each of 4 conditions); 24 foils (sentences that contained complement ambiguities that resolved to NP) and 48 fillers (sentences that did not contain complement ambiguity). Participants were rotated over one of four lists, implementing a Latin square design. 32 of the trials (including 8 of the experimental conditions) were followed immediately by a com-

prehension question. This was a simple question about the sentence immediately preceding that required the participant to make a yes or no response using the appropriate trigger button on the gamepad. The whole procedure took about 20 to 40 minutes, depending on the participant.

6 Results

This section shows how the comparisons were made between patterns of differential processing load at disambiguation in the parser metrics and the human metrics. Per-condition means of all metrics at the disambiguating word are given in Figure 3.

6.1 Regression path duration (RPD)

A linear mixed effects model (Bates et al., 2013) was constructed for regression path duration at the disambiguating word i.e., *walked* in the example sentences. RPD was modeled as a function of word length, word (unigram) frequency (Brants and Franz, 2006), ambiguity, and sentence type (type 1 is exemplified in sentence 1 and type 2 is exemplified in sentence 2), and the ambiguity x sentence type interaction; with random slopes for the ambiguity x sentence type interaction over both participant ID and over item ID. Word length and word frequency both exerted non-significant influences. There was a significant effect of ambiguity with the ambiguous conditions leading to 146 ms more RPD than the disambiguated conditions ($\beta = 135.15$, $SE = 37.60$, $t = 3.56$). There was a significant disadvantage for type 1 sentences of 79 ms as a main effect ($\beta = -68.59$, $SE = 30.66$, $t = -2.27$). There was significant interaction effect such that the effect of ambiguity in type 1 sentences was greater than the effect of ambiguity for type 2 sentences ($\beta = -64.28$, $SE = 31.33$, $t = -2.05$).

6.2 Phrase structure surprisal

Phrase structure surprisal predicted that the ambiguous cases would be harder then the unambiguous cases; and that the disadvantage of sentence type 1 in the ambiguous cases would turn around into a disadvantage of sentence type 2 in the unambiguous conditions. Individual terms for ambiguity and sentence type were included at each level of item. Effects of ambiguity, sentence type and the ambiguity x sentence type interaction were all significant in the model, and the shapes of these effects were broadly in line with the human data ($\beta = 0.65$, $SE = 0.05$, $t = 12.32$, $\beta = -0.11$, $SE = 0.03$, $t = -3.25$, and $\beta = -0.35$, $SE = 0.01$, $t = -62.35$ respectively).

6.3 Phrase structure entropy reduction

The directions of the entropy reduction hypothesis predictions were the same as for phrase structure surprisal, although there was a relatively greater difficulty with the type 2 cases versus surprisal. Effects of ambiguity, sentence type and the ambiguity x sentence type interaction were all significant in the model ($\beta = 0.32$, $SE = 0.02$, $t = 14.04$, $\beta = -0.03$, $SE = 0.02$, $t = -2.05$, and $\beta = -0.17$, $SE = 0.002$, $t = -55.79$ respectively). The shapes of these effects were broadly in line with the human data.

6.4 Dependency surprisal

The mean values of dependency surprisal at the disambiguating word show that ambiguous sentence types 1 and 2 are predicted to be equal. For the unambiguous cases, type 1 is predicted to be more difficult than type 2. Ambiguity did not exert a significant effect on dependency surprisal ($\beta = 0.0002$, $SE = 0.01$, $t = 0.01$). The effect of sentence type was significant, with type 1 causing more dependency surprisal than type 2 ($\beta = -0.09$, $SE = 0.01$, $t = -6.26$). The ambiguity x sentence type interaction was significant in the model ($\beta = 0.09$, $SE = 0.002$, $t = 39.67$) but the shape of the interaction did not match the shape of the human data: instead the model predicted a large effect of sentence type in the unambiguous conditions and a small effect of sentence type in the unambiguous control sentences.

6.5 Dependency retrieval time

The mean values for retrieval predicted that both of the ambiguous sentence types and unambiguous type 1 sentences should be equally difficult, with unambiguous type 1 predicted to cause the most difficulty. Main effects of ambiguity and sentence type were significant in the model ($\beta = -17.7$, $SE = 0.60$, $t = -29.72$ and $\beta = 17.7$, $SE = 0.6$, $t = 29.72$ respectively). There was a significant ambiguity x sentence type interaction ($\beta = -17.7$, $SE = 0.09$, $t = -191.25$). Comparing these prediction with the human data, the predictions are not in line with human performance at all.

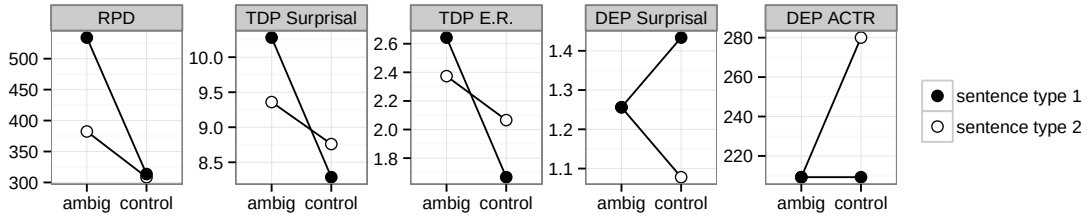

Figure 3: Per-condition means for each metric for the disambiguating word. RPD is the human eye movement measure *regression path duration*, see section 6.1. TDP Surprisal is surprisal computed over a phrase structure grammar, see section 6.2. TDP E.R. is entropy reduction computed over a phrase structure grammar, see section 6.3. DEP Surprisal is surprisal computed over a dependency grammar, section 6.4; DEP ATCR is retrieval time computed over a dependency grammar, section 6.5.

7 Conclusions

This section lays out the the conclusions that can be drawn from this work, paying attention to the question whether an information theoretic measure can be used in the NLG process as a proxy for human reading difficulty, as part of an effort to generate more readable texts.

For the metrics computed over a phrase structure grammar (phrase structure surprisal and phase structure entropy reduction), the comparison with human eye tracking metrics is relatively close. This suggests that phrase structure surprisal and phase structure entropy reduction are tracking human reading difficulty at disambiguation well. Phrase structure surprisal and phase structure entropy reduction are good predictors of the sort of human parsing difficulty that is measured by regression path duration, for these sentence types.

Dependency surprisal computed over a dependency grammar using a k-best parser with k=3 produces the wrong predictions for the complement ambiguity sentence types in this article. There is some scope for improving the predictions of this parser, as follows. Firstly setting k=3 may be restricting the beam width too much such that the ultimately-correct analysis is pruned too early. If so, simulations with increased values of k might be worth exploring. Secondly, one of the sentence types in this article relies on disambiguation by punctuation. Punctuation is well-handled in phrase structural grammars because it serves as a clause boundary marker, and phrase structure grammars natively express sentences as phrase combinations, whereas dependency grammars can only treat punctuation as a terminal in its own right. This might turn out to lead to an unfair comparison between dependency parser performance and phrase structure performance for the sentence types examined here. There is a clear case for examining dependency parsing for disambiguation types that use the sequence of words to effect disambiguation. Future work in this direction could take advantage of previous work with different ambiguities covered in e.g., Boston and Hale (2007) and Boston (2012), and extending it from using self-paced reading times to include eye-tracking metrics.

Dependency retrieval time did not show the interaction evident in the eye movement and phase grammar parser data. This suggests either that the Lewis and Vasishth (2005) model does not cover very well the sentence types used in this experiment, or that whatever coverage the Lewis and Vasishth (2005) model does have of the human data is obscured in the transformation from phrase structure grammar to dependency grammar versions of retrieval.

Previous work aimed at broad-coverage parsing evaluated against human eye movement corpora (Demberg and Keller, 2008; Boston et al., 2011) indicates that, in those corpus-derived linguistic environments, phrase structure surprisal and phase structure entropy reduction account for different components of variance in eye movement patterns. If future work continues to find that surprisal and entropy reduction predict human difficulty in psycholinguistic eye movement lab-based investigations (and the present paper shows how that can be done for one ambiguity type), then it will be reasonable to propose that a good model of sentence processing should use both surprisal and entropy reduction to predict (human) reading difficulty. Such a model would need to consider care-

fully the nature of the relationship between these different types of parser complexity. A starting point could be the observation that surprisal is essentially backwards-looking (seeks to disconfirm past analyses) whereas entropy reduction is essentially forward-looking (seeks to establish the uncertainty that remains at the current word with respect to how the rest of the sentence might pan out).

For NLG, the importance of this proposal is that such a model could be used to answer, algorithmically, questions that have previously only been satisfactorily answered in the laboratory. For example, in NLG the question often arises "For this proposition P, which we want the generator to put in a surface form SF for some given natural language L, which of the many possible SFs that express P in L should we produce?". So far this question has only been satisfactorily addressed by laboratory studies, which are few in number, expensive to run, and hard to generalise from.

When such generators are faced with this question, a better way forward would be to generate (some finite subset of) all possible SFs that express P in L, and then use surprisal and entropy reduction metrics as thresholds for pruning and ranking the SFs. This would lead the generator to produce only SFs that avoid syntactic complexity for the benefit of human readers. Different thresholds could produce texts tailor-made for groups with different reading abilities, or texts aimed to meet other constraints on acceptable human difficulty, e.g., texts for beginners learning a given natural language for the first time, or texts with different forms aimed at novices and experts.

Reiter and Belz (2009) discuss and evaluate some metrics for automatic evaluation of NLG in the context of generating weather forecasts. However these are designed to fit human measures at the whole-document level of NLG, different from the sentence-level incremental predictions generated and evaluated here. Also the evaluations discussed by those authors are done by fitting measures from offline human ratings of text readability, again different from the fine-grained detail of online human processing provided by the eye-tracking experiment here.

It seems clear that a combination of document-level and sentence-level predictors of human difficulty with generated text would be better than either alone for guiding NLG systems. It is conceivable that surprisal and entropy reduction might become useful automatic metrics for sentence-level evaluation of NLG texts, in the same way that BLEU (Papineni et al., 2002) and similar metrics serve in Machine Translation, but incrementally, and at a finer-grained and level.

References

J.R. Anderson. 2005. Human symbol manipulation within an integrated cognitive architecture. *Cognitive science*, 29(3):313–341.

Douglas Bates, Martin Maechler, Ben Bolker, and Steven Walker. 2013. lme4: Linear mixed-effects models using Eigen and S4. R package version 1.0-5.

M.F. Boston and J. Hale. 2007. Garden-pathing in a statistical dependency parser. In *Proceedings of the Midwest Computational Linguistics Colloquium*.

M.F. Boston, John T. Hale, Shravan Vasishth, and Reinhold Kliegl. 2011. Parallel processing and sentence comprehension difficulty. *Language and Cognitive Processes*, 26(3):301–349.

M.F. Boston. 2012. *A Computational Model of Cognitive Constraints in Syntactic Locality*. Ph.D. thesis, Cornell University, January.

M.F. Boston. 2013. Humdep3.0. An incremental dependency parser developed for human sentence processing modeling. `http://conf.ling.cornell.edu/Marisa`.

Thorsten Brants and Alex Franz. 2006. Web 1T 5-gram Version. computed from Google, published by Linguistic Data Consortium.

C. Clifton Jr. 1993. Thematic roles in sentence parsing. *Canadian Journal of Experimental Psychology*, 47(2):222–46.

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

F. Ferreira and J.M. Henderson. 1991. Recovery from misanalyses of garden-path sentences. *Journal of Memory and Language*, 30(6):725–745.

M.J. Green. 2014. *On Repairing Sentences: An Experimental and Computational Analysis of Recovery from Unexpected Syntactic Disambiguation in Sentence Parsing*. Ph.D. thesis, Psychology, Exeter.

J. 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 1–8, Morristown, NJ, USA. Association for Computational Linguistics.

J. Hale. 2004. The information-processing difficulty of incremental parsing. In F. Keller, S. Clark, M Crocker, and M. Steedman, editors, *ACL Workshop Incremental Parsing: Bringing Engineering and Cognition Together*, pages 58–65. Association for Computational Linguistics.

J. Hale. 2006. Uncertainty about the rest of the sentence. *Cognitive Science*, 30(4):643–672.

V.M. Holmes, A. Kennedy, and W.S. Murray. 1987. Syntactic structure and the garden path. *The Quarterly Journal of Experimental Psychology Section A: Human Experimental Psychology*, 39(2):2 – 277.

R. Levy. 2008. Expectation-Based Syntactic Comprehension. *Cognition*, 106(3):1126–1177.

R.L. Lewis and S. Vasishth. 2005. An activation-based model of sentence processing as skilled memory retrieval. *Cognitive Science*, 29:1–45.

B. McElree, S. Foraker, and L. Dyer. 2003. Memory structures that subserve sentence comprehension. *Journal of Memory and Language*, 48(1):67–91.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Umesh Patil, Shravan Vasishth, and Reinhold Kliegl. 2009. Compound effect of probabilistic disambiguation and memory retrievals on sentence processing: Evidence from an eyetracking corpus. In *Proceedings of 9th International Conference on Cognitive Modeling*, Manchester.

M.J. Pickering and M.J. Traxler. 1998. Plausibility and recovery from garden paths: An eye-tracking study. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 24(4):940–961.

K. Rayner and L. Frazier. 1987. Parsing temporarily ambiguous complements. *The Quarterly Journal of Experimental Psychology Section A: Human Experimental Psychology*, 39(4):657 – 673.

K. Rayner, A. Pollatsek, J. Ashby, and C. Clifton Jr. 2012. *Psychology of Reading*. Psychology Press, 2nd edition.

Ehud Reiter and Anja Belz. 2009. An investigation into the validity of some metrics for automatically evaluating natural language generation systems. *Computational Linguistics*, 35(4):529–558.

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

Brian Roark. 2004. Robust garden path parsing. *Natural language engineering*, 10(1):1–24.

B. Roark. 2013. tdparse. An incremental top down parser. `http://code.google.com/p/incremental-top-down-parser/`.

Claude Shannon. 1948. A mathematical theory of communication. *Bell Systems Technical Journal*, 27:379—423.

P. Sturt, M.J. Pickering, and M.W. Crocker. 1999. Structural Change and Reanalysis Difficulty in Language Comprehension. *Journal of Memory and Language*, 40:136–150.

J C Trueswell, M K Tanenhaus, and C Kello. 1993. Verb-specific constraints in sentence processing: separating effects of lexical preference from garden-paths. *J Exp Psychol Learn Mem Cogn*, 19(3):528–53.

S. Vasishth, S. Brüssow, R.L. Lewis, and H. Drenhaus. 2008. Processing polarity: How the ungrammatical intrudes on the grammatical. *Cognitive Science*, 32(4):685–712.