Predictability and compositionality

Jacob Louis Hoover[1,2], **Alessandro Sordoni**[3], **Timothy J. O'Donnell**[1,2,4]

draft 2020-06-08

[1]McGill University, Montréal, Canada
[2]Mila – Québec AI Institute, Montréal, Canada
[3]Microsoft Research, Montréal, Canada
[4]Canada CIFAR AI Chair, Mila

Abstract

Perhaps the most fundamental property of the system of human language is its underlying compositional structure. There is an intuitive connection between this grammatical structure and the statistics of word occurrences observed in language use. Work in language cognition relates the predictability of items in context to theories of language processing and acquisition. This intuitive connection is reflected also in NLP, in the assumption that the patterns of predictability correlate with linguistic structure. This assumption has been made explicitly in some approaches to unsupervised dependency parsing, and also present implicitly in the use of language modelling objectives for training modern neural models. The strongest version of this hypothesis is to say that compositional structure is in fact *entirely reducible* to cooccurrence statistics, a hypothesis made explicit in Futrell et al. (2019). Investigating the mutual information of pairs of words using pretrained contextualized embedding models, we show that the optimal structure for prediction is in fact not very closely correlated to the compositional structure. We propose that contextualized mutual information scores of this kind may be useful as a way to understand the structure of predictability, as a system distinct from compositional structure, but also integral to language use.

1 Introduction

Compositionality The celebrated creative capacity of human language derives fundamentally from its *compositional structure*. This property of linguistic structure, according to which meaningful expressions are built up from those of their component parts, is what allows language to encode an unlimited variety of novel meanings in a systematically generalizable way. In generative linguistics, the dominant view of natural language syntax is that it is the latent structure which describes how words are combined to construct an utterance and determine its well-formedness, as well as to derive its meaning from the meaning of its parts (e.g. Chomsky, 1957; Montague, 1970).

Under this view, syntactic structures can be seen as traces of the processes by which the meaning of a sentence is computed from the meanings of lexical items, with well-formedness judgments arising according to the satisfaction of constraints checked during the computation. There is widespread consensus in the field that these compositional processes are largely *lexically-driven*, meaning that the computations themselves are controlled by the individual words (or morphemes, or bundles of features) (e.g. Schabes, 1990; Steedman, 2000; Chomsky, 1995; Stabler, 1997). For example, a given verb may 'select for' a particular number of arguments (a subject, an object) and specify their features. This verb may be itself be selected for by a tense element, and so on, to build a sentence. This compositional hierarchical structure is often described in terms of a constituency tree in linguistics, but may alternatively be described in terms of a dependency graph, with selection establishing a relationship between a head and its dependent (see example in Figure 1).

1.1 Predictability v. compositional structure

There is an intuitive link between compositional structure and the cooccurrence patterns observable in language use. Grammatical rules will constrain the distribution of words in observed language use. Words in a dependency relationship with each other are words which fundamentally depend one upon the other, and cannot covary entirely independently. So, while syntactic dependencies as linguists define them are fundamentally about grammatical function and compositional structure, it is reasonable to expect that these structures will be linked to predictability and cooccurrence statistics.

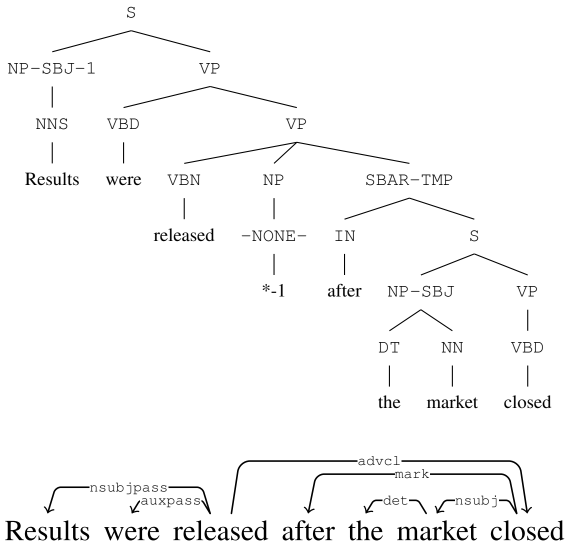

Figure 1: Example constituency tree annotation from the Penn Treebank, and the dependency structure derived from it. The translation from constituency tree to dependency tree is deterministic and largely, reversible. For the current purposes, it is important simply that in a broad sense the compositional structure of the sentence is represented by either formalism.

Predictability in processing and acquisition The predictability of linguistic items in context is widely assumed to play a role in language learning and language processing. In psycholinguistics, acquisition models which tie production and comprehension are based around an underlying prediction mechanism (e.g. Pickering and Garrod, 2013). Likewise, processing time has been shown to be directly related to predictability in context (e.g. Ehrlich and Rayner, 1981; Smith and Levy, 2008), leading to models in which context-derived expectation drives comprehension (see Levy, 2013). In particular, a notion of the predictability of one word based on the presence of another, termed *word association*, has been studied with respect to a number of different kinds of linguistic phenomena, based on semantic as well as syntactic constraints and lexical cooccurrence patterns (Palermo and Jenkins, 1964; Church and Hanks, 1990).

Unsupervised grammar induction Various methods have been used to measure the statistical covariance of pairs of words, with the intention of better understanding the how (or, the extent to which) the underlying structures of compositional syntax could be inferred from statistical facts about language data (see Terra and Clarke, 2003). In many early works on unsupervised dependency parsing (e.g., Van Der Mude and Walker, 1978; Magerman and Marcus, 1990; Yuret, 1998; Paskin, 2002) parse trees are inferred by finding the structures which optimize predictability by maximizing the conditional probability of dependants given their heads, or maximizing total pointwise mutual information between heads and dependants, two objectives which are equivalent (Mareček, 2012).

Contextualized word embeddings Recently, *contextualized word embedding* models have taken over the field of NLP. A contextualized word embedding is a map from words to vectors, but unlike traditional global word embeddings which map words to vectors independently as a function only of the word itself (the embedding function taking each word to a vector $w \mapsto \boldsymbol{h}$), contextualized word embeddings allow the representation of an individual word to be dependant on context (the embedding function takes sequences of words to sequences of vectors $(w_1, \dots, w_N) \mapsto (\boldsymbol{h}_1, \dots, \boldsymbol{h}_N)$). This design gives the potential for contextual information, including about syntactic relationships and presumably also compositional structure, to be learned and stored in the parameters which define this function.

These models are trained on a variety of *predictability-based* language modelling objectives (e.g., sentence generation, masked word prediction, next-sentence prediction, permutation language modelling; see Liu et al. (2020) for an overview). The use of contextual embedding models trained on large corpora has led to a jump in state-of-the-art performance for nearly every linguistic task to which they have been applied. Of particular note here are tasks that intuitively seem to implicate syntactic knowledge, such as question answering or recognizing textual entailment.

Predictability dependency hypothesis A natural question arises as to what degree predictability and syntax are correlated, or even if they are in effect the same thing, and one is reducible to the other. This is not a new question. A variation of the hypothesis that linguistic structure can be understood by understanding the way in which words influence each others' distributions is the motivation behind attempting unsupervised dependency parsing at all. In this paper we will refer to this idea loosely as the *predictability dependency hypothesis*. Linguists have traditionally pushed back against strong formulations of the predictability dependency hypothesis, saying that syntactic structure is fundamentally independent of cooccurrence statistics.[1] This argument is supported by the existence of sentences which are perfectly well-formed syntactically, but extremely unlikely statistically, the canonical illustration being the famous example sentence, *Colorless green ideas sleep furiously*, paired with its less-often cited partner *Furiously sleep ideas green colorless* (Chomsky, 1957). The first is syntactically valid, the second not, though both are supposed to be equally unlikely.

In this paper we make use of contextual embedding models to estimate a measure of mutual information between words given context, as a concrete predictability measure. With this measure we find evidence *against* the assumption that compositional structure can be recovered directly by optimizing for predictability. We begin to examine the ways in which the differences and similarities between compositional syntactic structures and structures optimized for predictability may be broken down in order to explain how these two kinds of information differ.

2 Background

2.1 Pointwise mutual information

We are interested in investigating the hypothesis that syntactic dependencies should be those which maximize predictability. To investigate this hypothesis we need to operationalize predictability in a concrete way. One common measure of predictability is *pointwise mutual information* (PMI), a symmetric function of the outcomes x, y of two random variables X, Y, defined as

$$\operatorname{pmi}(x;y) := \log \frac{p(x,y)}{p(x)p(y)} = \log \frac{p(x \mid y)}{p(x)}.$$

PMI is a measure of the amount of information about one outcome that is gained by learning the other.

With the random variables' outcomes being words, some estimate of PMI has been used in computational linguistic studies as a score of word association since Church and Hanks (1990).[2] PMI is useful as a measure of predictability since it is a measure of the change in information content. That is, it is a precise measure of how much *more* likely the outcomes are to occur together, compared to each occurring individually (for instance, it has the desirable property that even if the probability of two words occurring together is relatively low, if they even lower probability individually, then the PMI will be low).

The hypothesis that syntactic dependency structures correspond to trees which maximize mutual information between words has been widely, if often tacitly, assumed in work in NLP (de Paiva Alves, 1996; Yuret, 1998; Paskin, 2002; Klein and Manning, 2004).

2.2 Head–dependent mutual information hypothesis

Futrell et al. (2019) make the predictability dependency hypothesis explicit with their *Head–Dependent Mutual Information Hypothesis*, which proposes that syntactic dependencies are precisely the structures which maximize the expected value of PMI between heads and dependants. The intuition behind this hypothesis is that word pairs in dependency relationships are systematically constrained by the grammar, and therefore will control each other's distribution, statistically. They show empirically that mutual information between words linked by dependency is higher than that between non-dependent word pairs within the same sentence, matched for linear distance.

The hypothesis that syntactic dependency structures connect words in a way that maximizes mutual information leads to an important prediction if true: an unsupervised dependency parser should be derivable directly given an accurate estimator of PMI values between words. In this work we investigate this hypothesis making use of pretrained contextual embedding models to estimate PMI.

2.3 Contextualized embedding models and syntax

Having been trained on large amounts of language data, modern contextualized embedding networks seem to encode latent representations of syntactic structure to some extent, as explored in (Linzen et al., 2016; Blevins et al., 2018; Gulordava et al., 2018; Peters et al., 2018; Tenney et al., 2019), and in particular in Hewitt and Manning (2019), who used a linear probe to look for syntax in the latent representations of these models. This type of probe

[1] This line of argumentation may grant that cooccurrence statistics are predictable from syntax, if this correlation is only due to exogenous factors, such as nonlinguistic patterns in language data (deriving from facts about the world language is used to describe, rather than structural aspects of the linguistic system).

[2] They first extended the term *word association* from a more subjective meaning in the psycholinguistic literature, to refer to their statistical measure of covariance.

is refined in Voita and Titov (2020) who propose an information-theoretic version (which for this task estimates the amount of additional bits of information necessary to describe the syntactic structure given the pretrained model's representation), but the conclusion remains that a significant amount of the information needed to reconstruct dependency structures is stored within pretrained contextualized embeddings.

Our experiment is different from studies involving a supervised probe in two ways. First, we do not train a probe, so the map from latent representation to tree structure which we are interested in is arrived at in an unsupervised manner. Second, we are interested in comparing the structure of predictability to the compositional structure, so we are using these models as tools, not as experimental objects themselves. We use these pretrained networks as contextualized PMI estimators in order to recover the dependency structure which optimizes predictability.

2.4 Unsupervised dependency parsing as a test of the hypothesis

Simple estimates of pointwise mutual information based solely on word-frequency have been used for dependency parsing tasks (e.g., de Paiva Alves, 1996), and do not perform particularly well. However, a priori, this could be because these simple estimates miss the important information provided by context, not because cooccurrence is independent of compositional structure.

Below, using pretrained contextualized embedding models without fine-tuning, we extract estimates of pointwise mutual information between words and recover trees representing dependencies optimized for prediction. If predictability and compositional structure were reducible one to the other with syntactic dependency correlated with high PMI value, then a good PMI estimator will function as a syntactic dependency parser. We compare with hand-annotated gold dependencies, and find that structures which optimize mutual information do not correspond to a high degree with syntactic dependencies.

3 Method

3.1 A contextualized PMI score

Abstractly, the *conditional PMI* $\operatorname{pmi}(x; y \mid c)$ of the outcomes x, y of two random variables, given an outcome c of a third, is simply the PMI but with probabilities conditioned on c:

$$\operatorname{pmi}(x; y \mid c) = \log \frac{p(x, y \mid c)}{p(x|c)p(y|c)} = \log \frac{p(x \mid y, c)}{p(x \mid c)}.$$

In the context of measuring informativity between words in a sentence, the two observations are two words, and the conditioner is the rest of the sentence (without the two words). Neural models pretrained on language modelling objectives give a natural way to estimate this kind of conditional PMI between two words: simply compute the model's estimated log probability of the first word given the sentence (with that word masked), and subtract from this the log probability of this same word, but with the other word also masked.

A contextualized embedding model M gives us a function $p_M(w|c)$ for the probability assigned to word w given context c, so we can obtain a conditional pointwise mutual information estimate from M which we will call a **contextualized PMI** (CPMI) score, between two words $\mathbf{w}_I$ and $\mathbf{w}_J$ using a calculation of the form

$$\operatorname{cpmi}_{\mathrm{M}}(\mathbf{w}_I; \mathbf{w}_J \mid \mathbf{w}_{-I,J}) = \log \frac{p_M(\mathbf{w}_I \mid \mathbf{w}_{-I,J}, \mathbf{w}_J)}{p_M(\mathbf{w}_I \mid \mathbf{w}_{-I,J})} = \log \frac{p_M(\mathbf{w}_I \mid \mathbf{w}_{-I})}{p_M(\mathbf{w}_I \mid \mathbf{w}_{-I,J})}$$

where I and J are spans of indices, $\mathbf{w}_I$ is the set of subtokens with indices in I (likewise for $\mathbf{w}_J$), $\mathbf{w}_{-I}$ is the entire sentence without subtokens whose indices are in I, and $\mathbf{w}_{-I,J}$ is the sentence without subtokens whose indices are in I or J.

A note on subtokens We must formulate this measure in terms of sets of subtokens, rather than simply words, only because the models often break down the words into pieces smaller than words, for which gold dependencies are not defined.

To get an estimate for a probability $p(\mathbf{w})$ of a subtokenized word $\mathbf{w} = w_0, w_1, \ldots$ (that is to say, a joint probability, which we cannot get straight from a language model), we use a left-to-right chain rule decomposition of conditional probability estimates with the word:

$$p(\mathbf{w}) = p(w_0) \cdot p(w_1 \mid w_0) \cdot p(w_2 \mid w_0, w_1) \cdot \ldots$$

This decomposition allows us to estimate conditional pointwise information between words made of multiple subtokens, at the expense of specifying a left-to-right order within those words (see §3.2.1)

3.1.1 Global word embedding model

In addition to the CPMI estimates, we also compute a simpler non-contextualized PMI estimate from a global word embedding model, for comparison. We use Word2Vec (Mikolov et al., 2013)which maps a given word w_i in the vocabulary it to a 'target' embedding vector $\mathbf{w}_i$, as well as an 'context' embedding vector $\mathbf{c}_i$ (used during training). As demonstrated by Levy and Goldberg (2014); Allen and Hospedales (2019), Word2Vec's training objective is optimized when the inner product of the target and context embeddings equals the PMI, shifted by a global constant (determined by k, the number of negative samples): $\mathbf{w}_i^\top \mathbf{c}_j = \mathrm{pmi}(w_i; w_j) - \log k$.

This type of embedding model thus provides a non-contextual PMI estimator. A global shift will not change the resulting PMI-dependency trees, so we simply take $\mathrm{pmi}_{\mathrm{w2v}}(w_i; w_j) := \mathbf{w}_i^\top \mathbf{c}_j$, with embeddings calculated using a Word2Vec model trained on the same data as BERT.[3] This model will give a nonconditional PMI measure which will not take into account the positions of the words in a particular sentence, but should be able to reflect global distributional information similarly to the contextualized models, and should therefore function as a control with which to compare the PMI estimates derived from the contextualized models.

3.2 Extracting CPMI–dependency parses

Having computed a matrix of CPMI scores for a given sentence, we extract maximum-CPMI dependency structures by two methods. First, we extract a maximum spanning tree (MST; using Prim's algorithm, following Hewitt and Manning, 2019). This tree represents the optimal dependency structure for prediction according to the model. However, it will give dependency trees that may be non-projective, and since the gold trees extracted from the PTB are all projective, we also extract maximum projective spanning trees (using a dynamic programming algorithm from Eisner, 1996, 1997),[4] in order to be comparing similar structures. We should note that the imposition of such a projectivity constraint enforces that structures will look more like dependency structures. We must be careful about introducing too many such constraints if we are to interpret the resulting structure as purely being about predictability. However, the general accuracy of projective vs non-projective trees does not differ drastically (see below).

3.2.1 Symmetrizing matrices

PMI is a symmetric function, but the estimated CPMI scores from the contextual embedding models are not guaranteed to be symmetric, since nothing in the models' training explicitly forces their probability estimates to respect the identity $p(x|y)p(y) = p(y|x)p(x)$. For this reason, we have a choice when assigning a score to a pair of words $\mathbf{w}_I, \mathbf{w}_J$, whether we use the model's estimate of $\mathrm{cpmi}_{\mathrm{M}}(\mathbf{w}_I; \mathbf{w}_J)$ or of $\mathrm{cpmi}_{\mathrm{M}}(\mathbf{w}_J; \mathbf{w}_I)$. We calculate scores using both directions, and report their sum (likewise for the Word2Vec PMI estimate).

3.3 Dataset

We use gold dependencies for text from the Wall Street Journal, extracted from the Penn Treebank (PTB) corpus of English text hand-annotated for syntactic structure (Marcus et al., 1994; de Marneffe and Manning, 2008; de Marneffe et al., 2006). The PTB is annotated in the form of constituency parses. We convert these into basic Stanford Dependencies (de Marneffe et al., 2006).[5] The example given in Figure 1 shows a PTB constituency tree, and the dependency parse extracted from it.[6]

3.4 Contextualized embedding models

Using a representative subset of the WSJ data consisting of 1700 sentences (40117 words) from the standard development split as our corpus, for each sentence we compute CPMI scores for each pair of words using a number of pretrained contextualized embedding models (BERT (Devlin et al., 2019), XLNet (Yang et al., 2019), XLM (Lample and Conneau, 2019), BART (Lewis et al., 2019), DistilBERT (Sanh et al., 2019)).

[3] We use the implementation in *Gensim* (Řehůřek and Sojka, 2010), trained on BookCorpus and English Wikipedia, and use a global average vector for out-of-vocabulary words.

[4] Eisner's algorithm recovers the optimal projective *directed* dependency structure from a weighted ordered graph. Using a symmetric weight matrix, however, direction makes no difference, and we may treat the output dependency trees as undirected.

[5] For details on the dependency format, see the Stanford Dependencies manual (de Marneffe and Manning, 2008).

[6] It is worth noting that PTB trees are annotated according to rules which may seem somewhat outdated or nonstandard, from a linguistic perspective (for instance they are not binary branching). Likewise the dependencies do not make a distinction between arguments and adjuncts (by design; see Nivre et al., 2016; Przepiórkowski and Patejuk, 2018). For the current application however, we need only a broad and consistent annotation of syntactic structure, and these potential shortcomings should not be an issue.

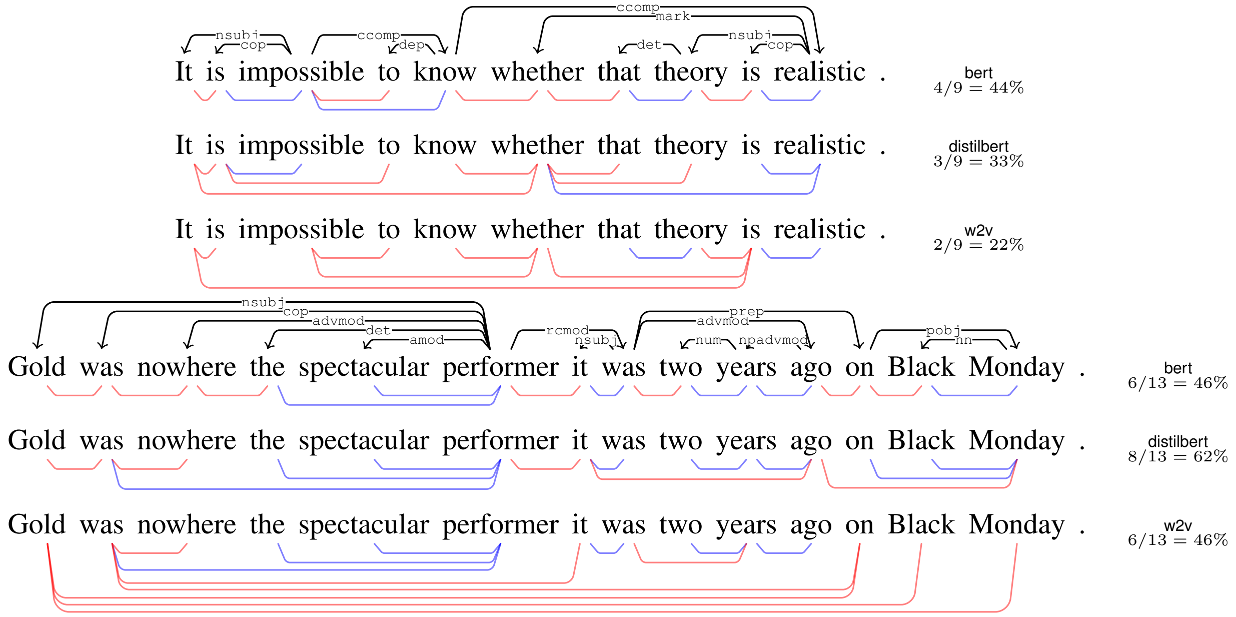
nsubj
cop
ccomp
dep
ccomp
mark
det
nsubj
cop
It is impossible to know whether that theory is realistic .
bert
4/9 = 44%
It is impossible to know whether that theory is realistic .
distilbert
3/9 = 33%
It is impossible to know whether that theory is realistic .
w2v
2/9 = 22%
nsubj
cop
advmod
det
amod
rcmod
nsubj
prep
advmod
num
npadvmod
pobj
nn
Gold was nowhere the spectacular performer it was two years ago on Black Monday .
bert
6/13 = 46%
Gold was nowhere the spectacular performer it was two years ago on Black Monday .
distilbert
8/13 = 62%
Gold was nowhere the spectacular performer it was two years ago on Black Monday .
w2v
6/13 = 46%

Figure 2: Projective parses for two example sentences, from BERT, DistilBERT, and the noncontextual baseline W2V. Gold dependency parse above in black, mutual information-based dependencies below, blue where they agree with gold dependencies, and red when they do not. The undirected unlabeled attachment score (UUAS) is printed at right (number of blue edges divided by total number of edges). Further examples in Figures 9, 10.

4 Results

Example CPMI-dependencies from two contextualized models and the noncontextualized word embedding model are given in Figure 2. For the purpose of illustration, further examples are given in Figures 9, 10 (at end). For the remainder of this section, we first look at the results in aggregate on a sentence-by-sentence level, and then in more detail, to examine the features which control the extent of the correlation.

4.1 Overall accuracy

Table 1 shows the mean accuracy of the CPMI-trees compared to the gold trees from the Penn Treebank. Accuracy here is calculated in terms of undirected unlabeled attachment score (UUAS), which is simply the number of edges which the CPMI structure and the gold dependency parse have in common, divided by the total number of edges in the sentence.[7] The numbers given in each row are the accuracy (UUAS), averaged across all sentences, for CPMI structures extracted from a given model as simple MSTs (left), or as MSTs with a projectivity constraint (right). Random baselines are obtained for each sentence by extracting a parse from a randomized weight matrix. The linear baseline parse is a tree in which the words are simply connected in linear order (corresponding to a degenerate left-to-right, or right-to-left dependency structure).

	MST	projective MST
random baseline	0.13	0.27
linear baseline	-	**0.50**
Word2Vec	0.31	0.41
Bart (large)	0.38	0.40
BERT (base cased)	0.46	0.47
BERT (large cased)	0.48	0.48
DistilBERT (cased)	0.48	0.49
XLNet (base cased)	0.44	0.47
XLNet (large cased)	0.39	0.43
XLM (MLM en 2048)	0.41	0.44

Table 1: Mean undirected accuracy scores for contextualized embedding models.

These results show broadly that CPMI-dependencies are not an accurate model of PTB dependencies. There is some variance with model, but in general, the average undirected accuracy score per sentence is a bit below 50%. While this is significantly above the random baselines, a simple linear baseline (degenerate left-to-right or right-to-left trees) corresponds more often to the gold

[7] Note that this accuracy score may be thought of as either precision or recall, since the number of gold edges and CPMI edges are the same.

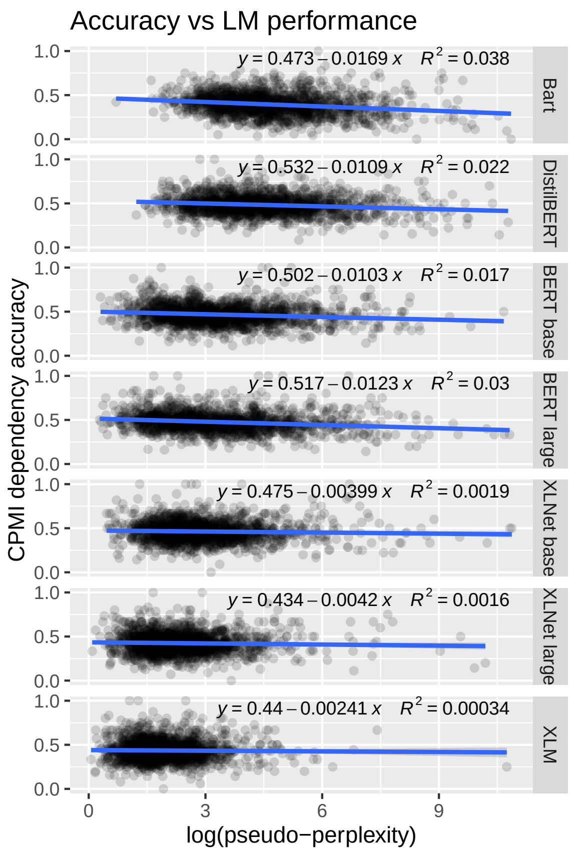

Figure 3: Per-sentence accuracy (UUAS) against log psuedo-perplexity. Accuracy is not tied to the confidence of the language model on a given sentence, for any of the models (there is a slight tendency to have higher accuracy on sentences of lower perplexity, but the effect size is negligible, and correlation is very low).

dependencies than any of the models' CPMI structures.

4.1.1 Accuracy versus perplexity

Even if the predictability dependency hypothesis were true in general, and dependency structures were highly correlated to the structures used for prediction, we might still expect a low mean accuracy for sentences on which the models were generally unsure in their predictions. If models confidence in predicting were tied to accuracy, then it might be harder to argue that the low accuracy score was due to the lack of correlation between mutual informativity and syntactic dependency, rather than to the models' struggling to recover such a structure. For this reason we briefly investigate the correlation between language model performance and CPMI-dependency accuracy.

Figure 3 shows that accuracy is *not* correlated with the models' confidence in their predictions, as measured in sentence-level perplexity. For this plot, model confidence is measured by obtaining a perplexity score for each sentence, calculated as the negative mean of the pseudo log-likelihood, that is,

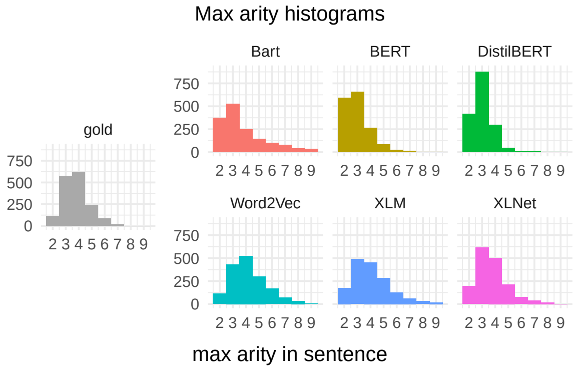

Figure 4: Histograms of max arity (note, histograms' tails are clipped at 10).

$$\text{pseudo PPL}(\mathbf{w}) = \exp\left[-\frac{1}{n}\sum_{I=1}^{n}\log p(\mathbf{w}_I|\mathbf{w}_{-I})\right]$$

where n is the length of sentence $\mathbf{w}$. The level of accuracy is not correlated to the confidence of the language model fitting a linear regression (R^2 below 0.04 for each of the models). The mean accuracy of CPMI-dependency edges is roughly the same on the sentences which the model predicts confidently (lower perplexity) as on the sentences which it predicts less confidently (higher perplexity).

We can conclude provisionally that structures optimized for prediction do not correspond well to compositional structures. However, consistent with the results reported in Futrell et al. (2019), neither are they entirely independent, (else accuracy would be closer to random), and these aggregate accuracy scores do not tell us much yet about what drives the correlation, to the extent to which it exists, nor in what way the predictability based dependency structures differ from syntactic compositional dependency structures.

4.1.2 Maximum arity

Examining the CPMI-dependency structures to understand in what ways they differ from the gold dependencies, we can first look at general shape of trees. One simple feature of the trees to look at is the arity, or valency the words: the number of other words with which they stand in a dependency relationship. Figure 4 shows histograms of the maximum arity per sentence for the different models.

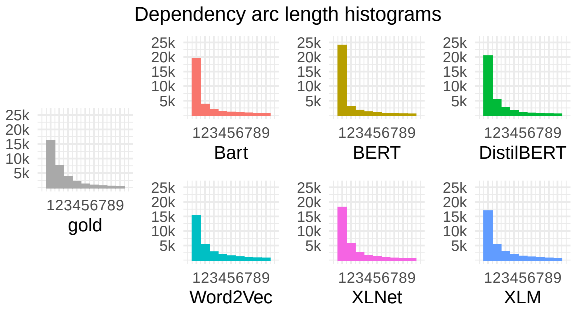

Figure 5: Histograms of arc length for the gold dependencies and CPMI dependencies from selected models. 48% of the gold arcs are length 1, whereas all of the CPMI-dependencies had a higher proportion. BERT's are 71% length 1, by far the most, compared with DistilBERT 60%, Bart 58%, XLNet 54%, XLM 50%. The noncontextualized baseline W2V was 39%. (Note, histograms' long tails are clipped at 10)

We can see that in general, for all the models, flat structures (that is, structures with maximum arity 2) are overrepresented compared to the gold (drastically so for BERT, DistilBERT, and Bart). Some models also have a higher number of sentences with max arity above 6, which are rare in the gold dependencies.

4.2 Examining features of the dependencies

To better understand the extent of the correlation between the CPMI dependencies and the gold dependencies, we examine the data we have, in terms of features of the individual dependency edges from the gold dependency structures (for a recall score) and the CPMI dependency structures (for a precision score). Even though the two kinds of structures agree only roughly half the time in aggregate, it is still possible that there is a stronger correlation for certain types of dependency arcs than for others.

We are interested in the extent to which the linguistic features of a particular dependency relationship (or of the pair of words involved) can explain when the CPMI-dependencies correspond to gold dependencies, and when they do not. To explore this correspondence we can look at the data thinking of accuracy as a binary dependent variable (that is, whether the two kinds of structure agree or do not agree), with observations being pairs of words in a given sentence which are in at least one or the other type of dependency relation. That is, we can view the corpus as a collection of edges of two kinds, gold dependencies, and CPMI dependencies. Our hypothesis is that there are certain features of dependencies which are more related to word prediction than others, and this analysis will help form an understanding of the manner and extent to which the abstract task of predicting words based on their contexts is related to the task of constructing syntactic parse.

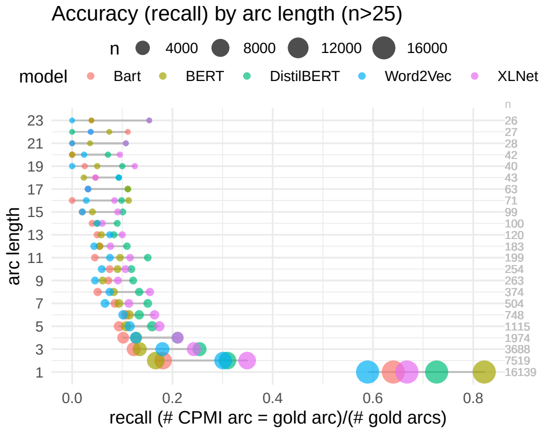

Figure 6: Edge-level recall accuracy, grouped by arc-length. Accuracy (plotted on the horizontal axis) is higher for shorter arcs. Note that the comparatively high accuracy of BERT's estimates overall seem to be driven by its predicting mostly arcs length 1. Interestingly this is less the case for DistilBERT, despite the similarity of the model.

4.2.1 Arc length

One simple feature to investigate is linear distance, or arc length (the distance between the pair of words connected by a dependency arc). Simple histograms of arc lengths are shown in Figure 5. One first observation is simply that short arcs are just more common. We are interested in determining whether the correlation between CPMI dependencies and syntactic dependencies is stronger for word-pairs that are closer to each other in the string. There is some difference in the distribution of lengths between the models, consistent with the exploration of max arity above (note the preponderance of length 1 arcs in BERT's estimates versus the others), but CPMI-dependencides from all models show a higher percentage of length 1 arcs than exist in the gold dependencies, so we expect a higher recall length 1 arcs (and a higher recall than precision), for all of the models.

Figure 6 shows the accuracy (recall) of CPMI-dependencies, grouping by arc length. There appears to be a roughly reciprocal relationship with lower recall for longer dependencies (this relationship holds for precision as well (not shown)). To simplify things slightly, we can perhaps just say

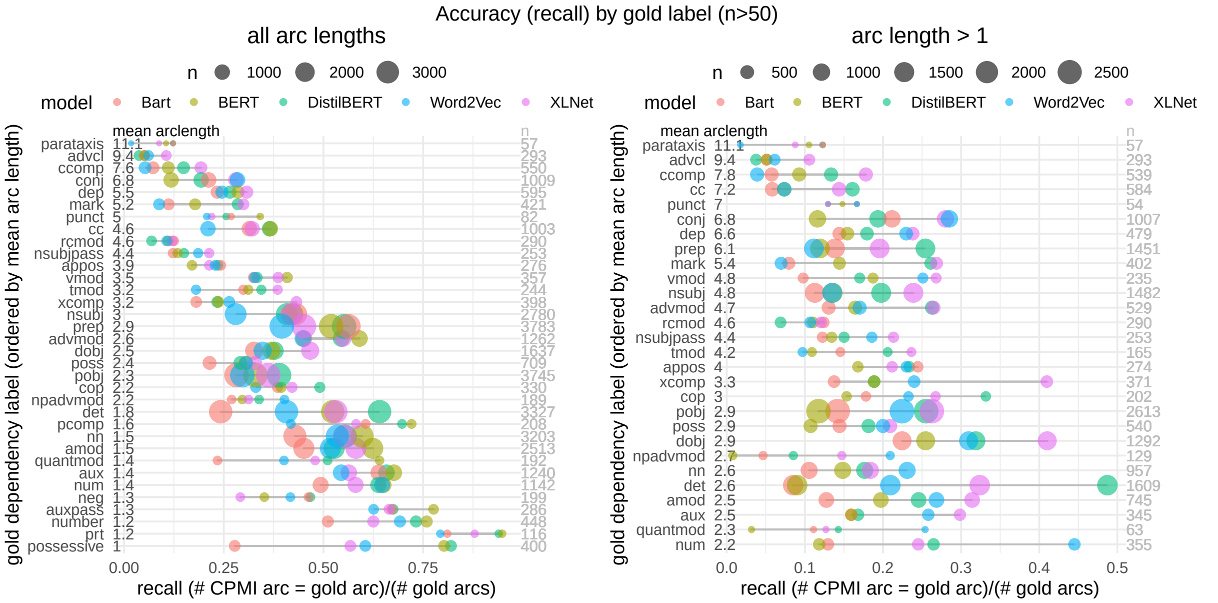

Figure 7: Plots of CPMI-dependency recall accuracy versus gold edge relation (on the vertical axis for readability, and ordered by mean arc length). Only dependency relations of which there are more than 50 observations are included. **Left**: Including dependency arcs of all lengths. **Right**: Including only arcs between nonadjacent words. Notice the correlation with mean length disappears when excluding the length 1 arcs.

more simply that there is a categorical distinction, with accuracy being higher for arcs of length 1 than for arcs of any length greater than 1.

4.2.2 Dependency relation label

One of the most obvious linguistic features that would theoretically be predictive of accuracy is the label of the gold dependency relation. That is, if our general hypothesis is correct, that there exist linguistically definable features by which the variation in accuracy may be explained, the type of dependency should at least partially explain this variation. This feature is only defined for word-pairs that are gold edges, so recall is the only accuracy measure which makes sense.[8]

In Figure 7, recall accuracy is plotted against relation label (for descriptions of labels in Stanford Dependencies de Marneffe and Manning, 2008). When examining all lengths of dependency together (left) relation seems to be correlated with accuracy; the CPMI accuracy of the different models roughly track each other in which relation types they are more accurate or less accurate for. Ordering the relations by their mean length reveals however that this effect is also closely related to arc length. The longer the mean length of a given relation the less likely it is to be predicted as a PMI dependency edge. Taking into account however the sharp difference in recall between length 1 and longer arcs from the exploration of the effect of length above (§4.2.1), we filter out the gold arcs of length 1 (this is roughly half of the total number of arcs, resulting new mean arclength numbers for each relation type). Then, looking at arcs length > 1, while there still may be an effect of relation type, it does not correlate with arc length (Figure 7, right).

4.3 Word classes

Further features to look at involve information about the specific words which are connected by the dependency arc. Here we investigate the effect of part of speech tags (from the gold annotation) which we have grouped by class, *open* vs. *closed* (roughly, content words v. function words).

In Figure 8 (at end), accuracy scores in terms of recall and precision are plotted against the classes of the two words connected by the dependency. As a rough way to inspect the effect of class independent of arc length, plots are given for dependencies of length 1 (adjacent words), and length > 1 (nonadjacent words), as well as for all lengths combined.

The clearest generalization that can be drawn initially is that the CPMI structures vastly overpredict closed-closed arcs, linking function words to

[8] Because, of course, pairs of words not in a dependency relation do not have a dependency relation label.

other function words many times more often than they are linked in the gold dependencies (in particular for nonadjacent word pairs). They also somewhat underpredict arcs linking two content words (open-open arcs). This observation is surprising in light of the a priori intuition that PMI structures should more often link open class words with other open class words, as Klein and Manning (2004) observe, “high mutual information between words is often stronger between two topically similar nouns than between, say, a preposition and its object.” In the study using noncontextualized word-frequency based PMI estimates that they are referring to, this seems to be the case; the noncontextualized PMI between two open-class words higher than that for closed-open pairs (like preposition-noun). However this is *not* what we see in our CPMI-structures (the predicted effect would look like higher recall for open-open pairs, and lower for closed-open pairs, which is not what we see). Beyond these observations, further patterns are hard to discern. Also, the noncontextualized PMI measure from Word2Vec does not seem to pattern very differently from the CPMI measures.

4.4 A note on structural constraints

In general, we have seen that predictability, as captured by the structures which maximize conditional PMI, does not immediately correspond to syntactic dependency. The predictability-based structures examined in detail here are recovered from the raw CPMI scores by extracting a maximum projective spanning tree, in order to compare to gold syntactic dependency structures representing compositional structure. The decision to extract a tree structure (rather than a more fully connected graph, or a full matrix) and further, a projective tree, is a constraint to increase the comparability of these two kinds of structure, and should magnify the correlation to the extent to which it exists. However, as briefly noted above, the introduction of a projectivity constraint did not drastically increase the accuracy scores. One might further consider introducing other structural constraints, in order to force the structure recovered from CPMI to have other properties in common with the gold dependencies. For instance biases used in unsupervised dependency parsing (e.g. Klein and Manning, 2004) such as introducing a distance kernel to the algorithm (making more long-distance connections more costly), or enforcing a restriction on valency (making it costly to connect a single word to many other words) would seem sensible steps toward maximizing the potential for overlap between the resulting structures and gold dependencies. However, the more such restrictions that we include, the less clearly we can interpret the resulting structure as being a representation of the structure used for prediction. (In addition, a distance kernel penalizing long arcs would likely not increase the accuracy scores, as the models are already predicting an over-large proportion length 1 dependencies, and likewise, a cursory examination of the distribution of valencies of the CPMI-structures is comparable to the distribution of valencies in the gold dependencies.)

5 Conclusion

In this study we have used a number of pretrained contextualized embedding models trained on prediction objectives to examine the optimal dependency-like structure describing prediction. We have found that this structure does not generally correspond well to compositional syntactic structure. To the extent that the pretrained embedding models are good estimators of predictability, this overall result gives evidence that the strong version of the predictability dependency hypothesis does not hold.

5.1 Interpretation

There are two important implications of this result. The first simply reinforces what might be concluded from earlier attempts to derive syntactic structure from corpus statistics using measures of mutual information. Using modern contextual embeddings with access to large amounts of data—models which have proven highly useful in NLP tasks—to extract structures of word-to-word connections which are most informative for prediction purposes, we see that those connections are not the same as the ones which are made in constructing a compositional derivation. This suggests that the poor performance of early models using PMI as an explicit way to do unsupervised dependency parsing (such as Yuret, 1998) cannot be blamed entirely on limitations inherent in using non-conditional PMI, or noncontextualized word representations.

The second point is due to the fact that while the prediction-based structures are not very good models of dependency parses, neither are they as bad as random baselines. There is a correlation, to some extent (as experimental results in Futrell et al. (2019) also suggested). For instance, the linear baseline which simply attaches each word to its immediate neighbours achieves slightly better accuracy than any of the models, and the prediction-based structures have a higher preference for connecting immediate neighbours than the gold dependency structures do. However, it is not the case that the two types of structure only correspond on length 1 dependencies. There is also some degree of correlation for longer dependencies.

5.2 Summary

Does compositional linguistic structure correspond to the structure of predictability? Using a measure of predictability defined in terms of conditional pointwise mutual information estimates obtained from contextual embedding models, we provide evidence that the answer is no in general. But, not entirely: the two types of structure are connected in ways not fully understood yet. Our preliminary results support the idea that these two types of information may exist in parallel partially-overlapping systems, and understanding them both may be helpful in designing models of language structure and use, and also may inform applications in NLP.

By definition, syntactic dependencies explicitly encode a specific kind of linguistic information. Modelling language data by optimizing purely for predictability may lead to important generalizations about the compositional structure of the underlying system being missed.

. . . .

Next steps

Retraining contextualized embedding models Recognizing that compositional dependency structure is not primarily what is used to do prediction prompts further investigation to understand how the two structures systematically differ. One initial goal with this study is to discover how to best integrate the two kinds of information for downstream tasks. Initially, this will include experiments with retraining contextual embedding models to be better estimates of mutual information (including for instance, regularization to force model to learn probability estimates which respect the identity $p(x|y)p(y) = p(y|x)p(x)$), to derive a more clearly interpretable estimate of the optimal structure of the information used for prediction. This estimate itself may be useful as an input feature for parsing tasks.

CPMI and syntactic contrasts Despite the demonstrated lack of evidence for the strong predictability dependency hypothesis, it remains true that syntactic structure affects cooccurrence statistics. For example, the subject position of a verb with plural morphology is vastly more likely to be occupied by a plural noun than a singular. Our CPMI measure may be applied to test the extent to which this kind of knowledge is represented by the contextual embedding models (is the PMI between the agreeing forms reliably higher than the PMI between non-agreeing forms?). A corpus of syntactic example sets will provide a starting place for experimentation. A cursory initial exploration of agreement across relative clauses using the SyntaxGym dataset (Gauthier et al., 2020; Hu et al., 2020) seems to show that the presence/absence of agreement has little to no effect on the mutual information, but a more thorough exploration is warranted.

Factorizing the predictability structure There is a larger hypothesis to explore concerning what we can expect to be described by probabilistic models of human language. It seems likely that compositional dependency structure is in some sense useful for word prediction, and in a systematic way. However, as the current study has shown, optimizing dependencies for prediction does not recover the compositional structure, but instead a structure which is quite different, without being entirely unrelated. This partially-overlapping relationship suggests that the true probability structure may be factorizable into components. Syntactic structure may control one component, while other communicative and real-world effects, such as lexical semantics and patterns of language use, controls another component. Further investigation in this direction is a longer term goal.

References

Carl Allen and Timothy Hospedales. 2019. Analogies explained: Towards understanding word embeddings.

Terra Blevins, Omer Levy, and Luke Zettlemoyer. 2018. Deep rnns encode soft hierarchical syntax. *arXiv preprint arXiv:1805.04218*.

Noam Chomsky. 1957. *Syntactic Structures*. Mouton and Co., The Hague.

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

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1):22–29.

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

Susan F Ehrlich and Keith Rayner. 1981. Contextual effects on word perception and eye movements during reading. *Journal of Memory and Language*, 20(6):641.

Jason Eisner. 1996. Three new probabilistic models for dependency parsing: An exploration. In *Proceedings of the 16th conference on Computational linguistics-Volume 1*, pages 340–345. Association for Computational Linguistics.

Jason Eisner. 1997. An empirical comparison of probability models for dependency grammar.

Richard Futrell, Peng Qian, Edward Gibson, Evelina Fedorenko, and Idan Blank. 2019. Syntactic dependencies correspond to word pairs with high mutual information. In *Proceedings of the fifth international conference on dependency linguistics (depling, syntaxfest 2019)*, pages 3–13.

Jon Gauthier, Jennifer Hu, Ethan Wilcox, Peng Qian, and Roger Levy. 2020. SyntaxGym: An online platform for targeted evaluation of language models. In *Proceedings of the Association for Computational Linguistics: System Demonstrations (ACL 2020)*.

Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. 2018. Colorless green recurrent networks dream hierarchically. In *Proceedings of NAACL-HLT 2018*, pages 1195–1205, New Orleans, Louisiana. Association for Computational Linguistics.

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

Jennifer Hu, Jon Gauthier, Peng Qian, Ethan Wilcox, and Roger P. Levy. 2020. A systematic assessment of syntactic generalization in neural language models.

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

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining.

Omer Levy and Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2177–2185. Curran Associates, Inc.

Roger Levy. 2013. Memory and surprisal in human sentence comprehension. Corrected verison, 2015.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the ability of lstms to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535.

Qi Liu, Matt J. Kusner, and Phil Blunsom. 2020. A survey on contextual embeddings.

David M Magerman and Mitchell P Marcus. 1990. Parsing a natural language using mutual information statistics. In *AAAI*, volume 90, pages 984–989.

Mitchell Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The Penn Treebank: Annotating predicate argument structure. In *Proceedings of the Workshop on Human Language Technology*, HLT '94, pages 114–119, USA. Association for Computational Linguistics.

David Mareček. 2012. *Unsupervised Dependency Parsing*. MFF UK, Prague, Czech Republic. PhD Thesis.

Marie-Catherine de Marneffe, Bill MacCartney, and Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy. European Language Resources Association (ELRA).

Marie-Catherine de Marneffe and Christopher Manning. 2008. *Stanford typed dependencies manual*. Stanford NLP.

Marie-Catherine de Marneffe and Christopher D. Manning. 2008. The stanford typed dependencies representation. In *COLING 2008 Workshop on Cross-framework and Cross-domain Parser Evaluation*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Richard Montague. 1970. Universal grammar. *Theoria*, 36(3):373–398.

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

Eduardo de Paiva Alves. 1996. The selection of the most probable dependency structure in japanese using mutual information. In *34th Annual Meeting of the Association for Computational Linguistics*, pages 372–374.

David S Palermo and James J Jenkins. 1964. Word association norms.

Mark A Paskin. 2002. Grammatical bigrams. In *Advances in neural information processing systems*, pages 91–97.

Matthew E Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018. Dissecting contextual word embeddings: Architecture and representation. *arXiv preprint arXiv:1808.08949*.

Martin J. Pickering and Simon Garrod. 2013. An integrated theory of language production and comprehension. *Behavioral and Brain Sciences*, 36(4):329–347.

Adam Przepiórkowski and Agnieszka Patejuk. 2018. Arguments and adjuncts in universal dependencies. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3837–3852.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.

Yves Schabes. 1990. *Mathematical and computational aspects of lexicalized grammars*. Ph.D. thesis, University of Pennsylvania.

Nathaniel J Smith and Roger Levy. 2008. Probabilistic prediction and the continuity of language comprehension. In *9th Conference on Conceptual Structure, Discourse, and Language (CSDL9)*.

Edward Stabler. 1997. Derivational minimalism. In *Logical Aspects of Computational Linguistics*, pages 68–95, Berlin, Heidelberg. Springer Berlin Heidelberg.

Mark Steedman. 2000. The syntactic process.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R Bowman, Dipanjan Das, et al. 2019. What do you learn from context? probing for sentence structure in contextualized word representations. *arXiv preprint arXiv:1905.06316*.

Egidio Terra and C. L. A. Clarke. 2003. Frequency estimates for statistical word similarity measures. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL ’03, pages 165–172, USA. Association for Computational Linguistics.

Antony Van Der Mude and Adrian Walker. 1978. On the inference of stochastic regular grammars. *Information and Control*, 38(3):310–329.

Elena Voita and Ivan Titov. 2020. Information-theoretic probing with minimum description length.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding.

Deniz Yuret. 1998. *Discovery of Linguistic Relations Using Lexical Attraction*. Ph.D. thesis, Massachusetts Institute of Technology, Dept. of Electrical Engineering and Computer Science.

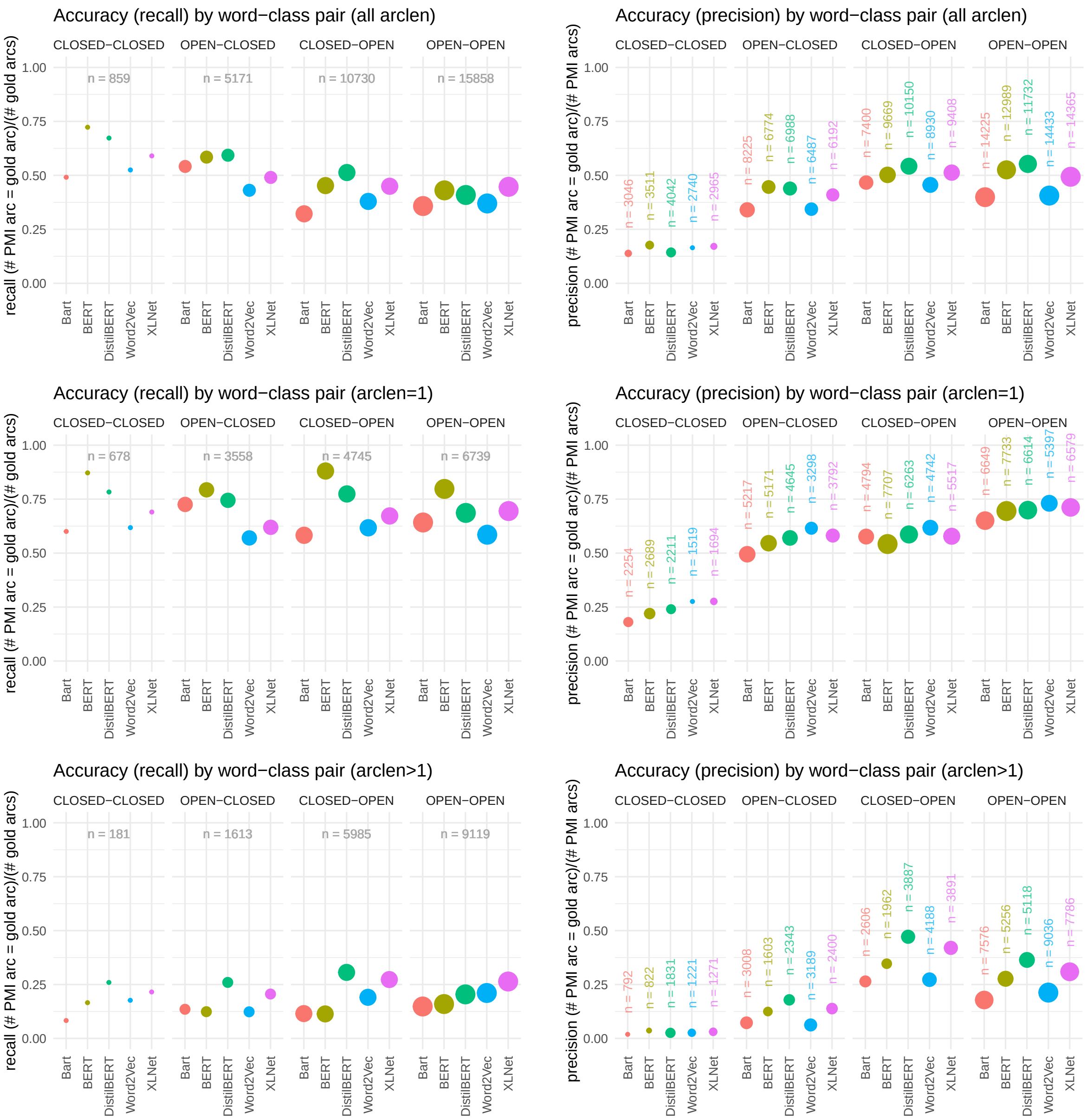

Figure 8: Accuracy by type of word-class pair, in terms of recall (left) and precision (right), looking at all arcs (top), only arcs of length 1 (middle), or all arcs of length greater than 1 (bottom). Vertical axis is accuracy score; points are labelled with the number of arcs of the given type. While there may be some more nuanced patterns to be found, the overall conclusions to be drawn is that arcs of length 1 are more accurate, but are overrepresented in the CPMI dependencies, compared to the gold (so, higher recall, lower precision). We also do not see a large difference between the CPMI estimates and the PMI estimates from Word2Vec.

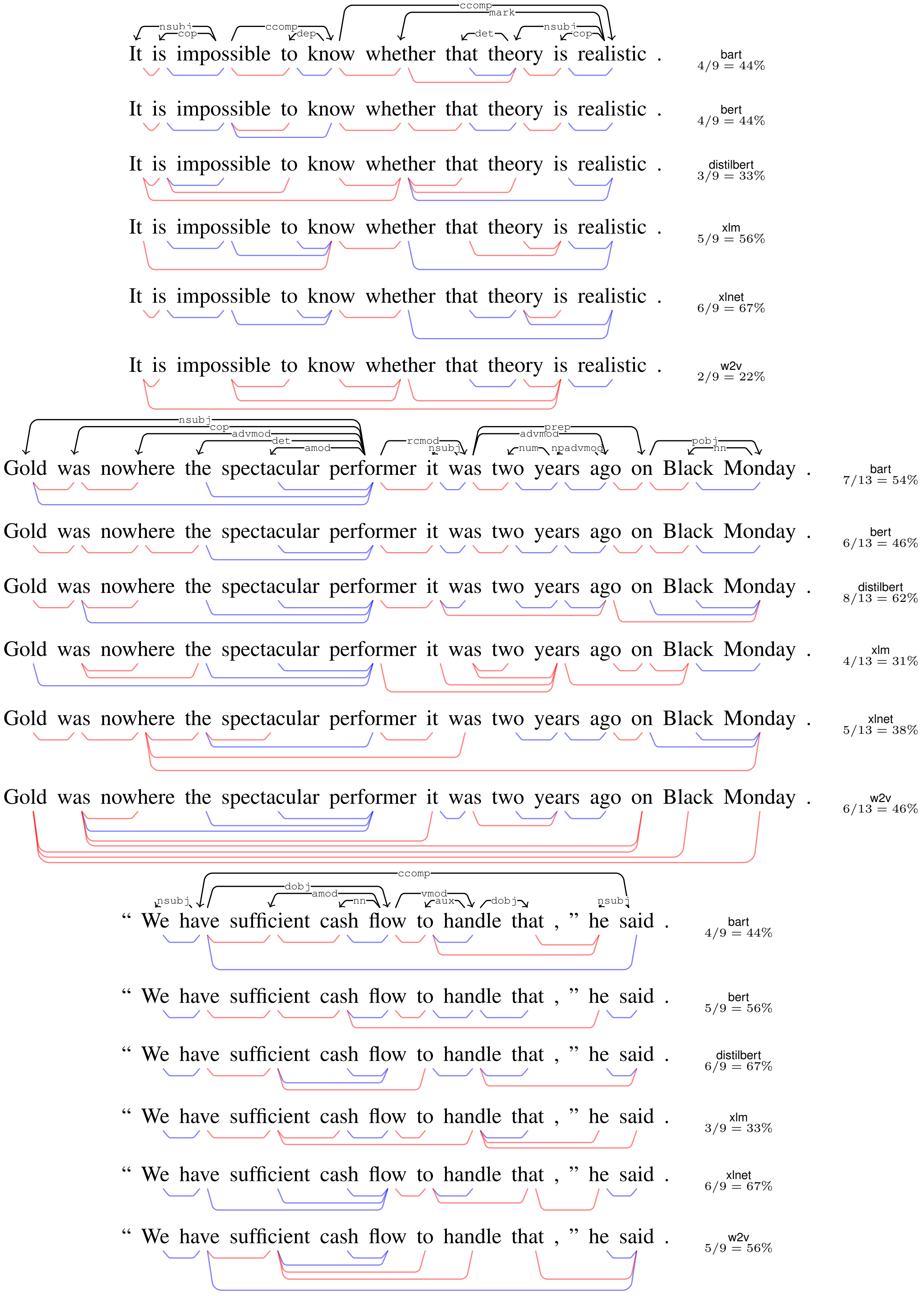
nsubj
cop
ccomp
dep
ccomp
mark
det
nsubj
cop
It is impossible to know whether that theory is realistic .
bart
4/9 = 44%
It is impossible to know whether that theory is realistic .
bert
4/9 = 44%
It is impossible to know whether that theory is realistic .
distilbert
3/9 = 33%
It is impossible to know whether that theory is realistic .
xlm
5/9 = 56%
It is impossible to know whether that theory is realistic .
xlnet
6/9 = 67%
It is impossible to know whether that theory is realistic .
w2v
2/9 = 22%
nsubj
cop
advmod
det
amod
rcmod
nsubj
prep
advmod
num
npadvmod
pobj
nn
Gold was nowhere the spectacular performer it was two years ago on Black Monday .
bart
7/13 = 54%
Gold was nowhere the spectacular performer it was two years ago on Black Monday .
bert
6/13 = 46%
Gold was nowhere the spectacular performer it was two years ago on Black Monday .
distilbert
8/13 = 62%
Gold was nowhere the spectacular performer it was two years ago on Black Monday .
xlm
4/13 = 31%
Gold was nowhere the spectacular performer it was two years ago on Black Monday .
xlnet
5/13 = 38%
Gold was nowhere the spectacular performer it was two years ago on Black Monday .
w2v
6/13 = 46%
ccomp
dobj
amod
nn
nsubj
vmod
aux
dobj
nsubj
“ We have sufficient cash flow to handle that , ” he said .
bart
4/9 = 44%
“ We have sufficient cash flow to handle that , ” he said .
bert
5/9 = 56%
“ We have sufficient cash flow to handle that , ” he said .
distilbert
6/9 = 67%
“ We have sufficient cash flow to handle that , ” he said .
xlm
3/9 = 33%
“ We have sufficient cash flow to handle that , ” he said .
xlnet
6/9 = 67%
“ We have sufficient cash flow to handle that , ” he said .
w2v
5/9 = 56%

Figure 9: Examples of projective parses from Bart, BERT, DistilBERT, XLM, XLNet, and the noncontextual baseline W2V. Gold dependency parse above in black, mutual information-based dependencies below, blue where they agree with gold dependencies, and red when they do not. Accuracy scores (UUAS) are given for each sentence.

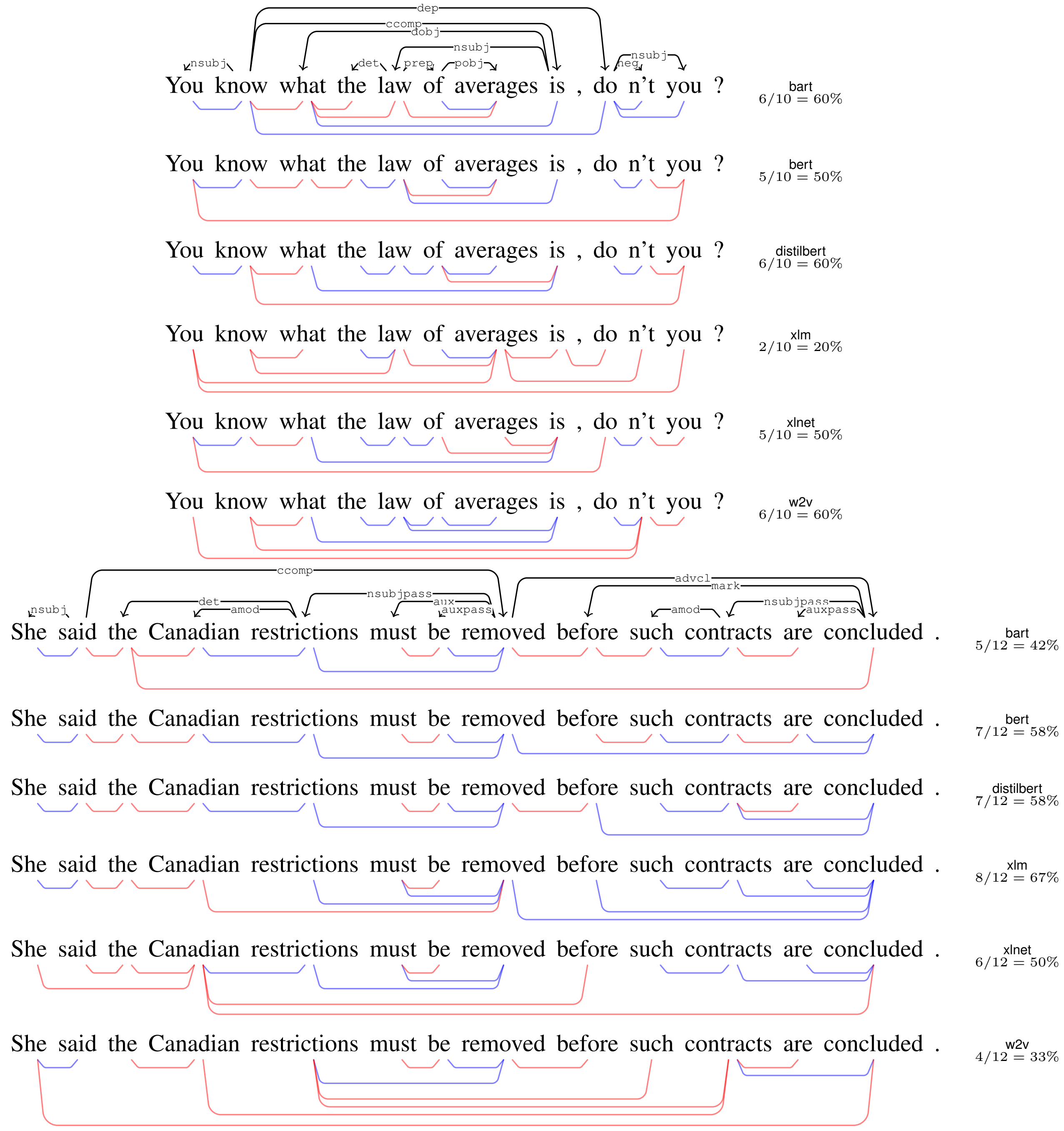

Figure 10: Further examples of projective parses from Bart, BERT, DistilBERT, XLM, XLNet, and the noncontextual baseline W2V.