Emergence of Syntax Needs Minimal Supervision

Raphaël Bailly, Kata Gábor

Emergence of Syntax Needs Minimal Supervision

Raphaël Bailly
SAMM, EA 4543, FP2M 2036 CNRS
Université Paris 1 Panthon-Sorbonne
raphael.bailly@univ-paris1.fr

Kata Gábor
ERTIM, EA 2520
INALCO
kata.gabor@inalco.fr

Abstract

This paper is a theoretical contribution to the debate on the learnability of syntax from a corpus without explicit syntax-specific guidance. Our approach originates in the observable structure of a corpus, which we use to define and isolate grammaticality (syntactic information) and meaning/pragmatics information. We describe the formal characteristics of an autonomous syntax and show that it becomes possible to search for syntax-based lexical categories with a simple optimization process, without any prior hypothesis on the form of the model.

1 Introduction

Syntax is the essence of human linguistic capacity that makes it possible to produce and understand a potentially infinite number of unheard sentences. The principle of compositionality (Frege, 1892) states that the meaning of a complex expression is fully determined by the meanings of its constituents *and its structure*; hence, our understanding of sentences we have never heard before comes from the ability to construct the sense of a sentence out of its parts. The number of constituents and assigned meanings is necessarily finite. Syntax is responsible for creatively combining them, and it is commonly assumed that syntax operates by means of algebraic compositional rules (Chomsky, 1957) and a finite number of syntactic categories.

One would also expect a computational model of language to have - or be able to acquire - this compositional capacity. The recent success of neural network based language models on several NLP tasks, together with their "black box" nature, attracted attention to at least two questions. First, when recurrent neural language models generalize to unseen data, does it imply that they acquire syntactic knowledge, and if so, does it translate into human-like compositional capacities (Baroni, 2019; Lake and Baroni, 2017; Linzen et al., 2016; Gulordava et al., 2018)? Second, whether research into neural networks and linguistics can benefit each other (Pater, 2019; Berent and Marcus, 2019); by providing evidence that syntax can be learnt in an unsupervised fashion (Blevins et al., 2018), or the opposite, humans and machines alike need innate constraints on the hypothesis space (a universal grammar) (Adhiguna et al., 2018; van Schijndel et al., 2019)?

A closely related question is whether it is possible to learn a language's syntax exclusively from a corpus. The *poverty of stimulus* argument (Chomsky, 1980) suggests that humans cannot acquire their target language from only positive evidence unless some of their linguistic knowledge is innate. The machine learning equivalent of this categorical "no" is a formulation known as *Gold's theorem* (Gold, 1967), which suggests that the complete unsupervised learning of a language (correct grammaticality judgments for every sequence), is intractable from only positive data. Clark and Lappin (2010) argue that Gold's paradigm does not resemble a child's learning situation and there exist algorithms that can learn unconstrained classes of infinite languages (Clark and Eyraud, 2006). This ongoing debate on syntax learnability and the poverty of the stimulus can benefit from empirical and theoretical machine learning contributions (Lappin and Shieber, 2007; McCoy et al., 2018; Linzen, 2019).

In this paper, we argue that syntax can be inferred from a sample of natural language with very minimal supervision. We introduce an information theoretical definition of what constitutes syntactic information. The linguistic basis of our approach is the autonomy of syntax, which we redefine in terms of (statistical) independence. We demonstrate that it is possible to establish a syntax-based lexical classification of words from a corpus without a prior hypothesis on the form of a syntactic model.

Our work is loosely related to previous attempts at optimizing language models for syntactic performance (Dyer et al., 2016; Adhiguna et al., 2018) and more particularly to Li and Eisner (2019) because of their use of mutual information and the information bottleneck principle (Tishby et al., 1999). However, our goal is different in that we demonstrate that very minimal supervision is sufficient in order to guide a symbolic or statistical learner towards grammatical competence.

2 Language models and syntax

As recurrent neural network based language models started to achieve good performance on different tasks (Mikolov et al., 2010), this success sparked attention on whether such models implicitly learn syntactic information. Language models are typically evaluated using perplexity on test data that is similar to the training examples. However, lower perplexity does not necessarily imply better *syntactic* generalization. Therefore, new tests have been put forward to evaluate the linguistically meaningful knowledge acquired by LMs.

A number of tests based on artificial data have been used to detect compositionality or systematicity in deep neural networks. Lake and Baroni (2017) created a task set that requires executing commands expressed in a compositional language. Bowman et al. (2015) design a task of logical entailment relations to be solved by discovering a recursive compositional structure. Saxton et al. (2019) propose a semi-artificial probing task of mathematics problems.

Linzen et al. (2016) initiated a different line of linguistically motivated evaluation of RNNs. Their data set consists in minimal pairs that differ in grammaticality and instantiate sentences with long distance dependencies (e.g. number agreement). The model is supposed to give a higher probability to the grammatical sentence. The test aims to detect whether the model can solve the task even when this requires knowledge of a hierarchical structure. Subsequently, several alternative tasks were created along the same concept to overcome specific shortcomings (Bernardy and Lappin, 2017; Gulordava et al., 2018), or to extend the scope to different languages or phenomena (Ravfogel et al., 2018, 2019).

It was also suggested that the information content of a network can be tested using ”probing tasks” or ”diagnostic classifiers” (Giulianelli et al., 2018; Hupkes et al., 2018). This approach consists in extracting a representation from a NN and using it as input for a supervised classifier to solve a different linguistic task. Accordingly, probes were conceived to test if the model learned parts of speech (Saphra and Lopez, 2018), morphology (Belinkov et al., 2017; Peters et al., 2018a), or syntactic information. Tenney et al. (2019) evaluate contextualized word representations on syntactic and semantic sequence labeling tasks. Syntactic knowledge can be tested by extracting constituency trees from a network's hidden states (Peters et al., 2018b) or from its word representations (Hewitt and Manning, 2019). Other syntactic probe sets include the work of Conneau et al. (2018) and Marvin and Linzen (2018).

Despite the vivid interest for the topic, no consensus seems to unfold from the experimental results. Two competing opinions emerge:

- Deep neural language models generalize by learning human-like syntax: given sufficient amount of training data, RNN models approximate human compositional skills and implicitly encode hierarchical structure at some level of the network. This conjecture coincides with the findings of, among others Bowman et al. (2015); Linzen et al. (2016); Giulianelli et al. (2018); Gulordava et al. (2018); Adhiguna et al. (2018).

- The language model training objective does not allow to learn compositional syntax from a corpus alone, no matter what amount of training data the model was exposed to. Syntax learning can only be achieved with task-specific guidance, either as explicit supervision, or by restricting the hypothesis space to hierarchically structured models (Dyer et al., 2016; Marvin and Linzen, 2018; Chowdhury and Zamparelli, 2018; van Schijndel et al., 2019; Lake and Baroni, 2017).

Moreover, some shortcomings of the above probing methods make it more difficult to come to a conclusion. Namely, it is not trivial to come up with minimal pairs of naturally occurring sentences that are equally likely. Furthermore, assigning a (slightly) higher probability to one sentence does not reflect the nature of knowledge behind a grammaticality judgment. Diagnostic classifiers may do well on a linguistic task because they learn to solve it, not because their input contains a hierarchical structure (Hewitt and Liang, 2019). In what follows, we present our assessment on how the difficulty of creating a linguistic probing data set is interconnected with the theoretical problem of learning a model of syntactic competence.

2.1 Competence or performance, or why syntax drowns in the corpus

If syntax is an autonomous module of linguistic capacity, the rules and principles that govern it are formulated independently of meaning. However, a corpus is a product of language use or *performance*. Syntax constitutes only a subset of the rules that generate such a product; the others include communicative needs and pragmatics. Just as meaning is uncorrelated with grammaticality, corpus frequency is only remotely correlated with human grammaticality judgment (Newmeyer, 2003).

Language models learn a probability distribution over sequences of words. The training objective is not designed to distinguish grammatical from agrammatical, but to predict language use. While Linzen et al. (2016) found a correlation between the perplexity of RNN language models and their syntactic knowledge, subsequent studies (Bernardy and Lappin, 2017; Gulordava et al., 2018) recognized that this result could have been achieved by encoding lexical semantic information, such as argument typicality. E.g. ”in ’*dogs (...) bark*’, an RNN might get the right agreement by encoding information about what typically barks” (Gulordava et al., 2018).

Several papers revealed the tendency of deep neural networks to fixate on surface cues and heuristics instead of ”deep” generalization in solving NLP tasks (Levy et al., 2015; Niven and Kao, 2019). In particular, McCoy et al. (2019) identify three types of syntactic heuristics that get in the way of meaningful generalization in language models.

Finally, it is difficult to build a natural language data set without semantic cues. Results from the syntax-semantics interface research show that lexical semantic properties account for part of syntactic realization (Levin and Rappaport Hovav, 2005).

3 What is syntax a generalization of?

We have seen in section 2 that previous works on the linguistic capacity of neural language models concentrate on compositionality, the key to creative use of language. However, this creativity is not present in language models: they are bound by the type of the data they are exposed to in learning.

We suggest that it is still possible to learn syntactic generalization from a corpus, but not with likelihood maximization. We propose to isolate the syntactic information from shallow performance-related information. In order to identify such information without explicitly injecting it as direct supervision or model-dependent linguistic presuppositions, we propose to examine *inherent structural properties* of corpora. As an illustration, consider the following natural language sample:

cats eat rats
rats fear cats
mathematicians prove theorems
doctors heal wounds

According to the Chomskyan principle of the *autonomy of syntax* (Chomsky, 1957), the syntactic rules that define well-formedness can be formulated without reference to meaning and pragmatics. For instance, the sentence *Colorless green ideas sleep furiously* is grammatical for humans, despite being meaningless and unlikely to occur. We study whether it is possible to deduce, from the *structural properties* of our sample above, human-like grammaticality judgments that predict sequences like *cats rats fear* as agrammatical, and accept e.g. *wounds eat theorems* as grammatical.

We distinguish two levels of observable structure in a corpus:

1. the proximity; the tendency of words to occur in the context of each other (in the same document/same sentence, etc.)

2. the order in which the words appear.

Definition 1. *Let L be a language over vocabulary V. The language that contains every possible sequence obtained by shuffling the elements in a sequence of L will be denoted $\overline{L}$.*

If V^* is the set of every possible sequence over vocabulary V and L is the language instantiated by our corpus, L is generated by a mixture of contextual and syntactic constraints over V^*. We are looking to separate the syntactic specificities from the grammatically irrelevant, contextual cues. The processes that transform V^* into $\overline{L}$, and $\overline{L}$ into L

$$V^* \xrightarrow{\text{proximity}} \overline{L} \xrightarrow{\text{order}} L$$

are entirely dependent on words: it should be possible to encode the information used by these processes into word categories.

In what follows, we will provide tools to isolate the information involved in proximity from the information involved in order. We also relate these categories to linguistically relevant concepts.

3.1 Isolating syntactic information

For a given word, we want to identify the information involved in each type of structure of the corpus, and represent it as partitions of the vocabulary into lexical categories:

1. **Contextual** information is any information unrelated to sentence structure, and hence, grammaticality: this encompasses meaning, topic, pragmatics, corpus artefacts etc. The surface realization of sentence structure is a language-specific combination of word order and morphological markers.

2. **Syntactic** information is the information related to sentence structure and - as for the autonomy requirement - nothing else: it is independent of all contextual information.

In the rest of the paper we will concentrate on English as an example, a language in which syntactic information is primarily encoded in order. In section 5 we present our ideas on how to deal with morphologically richer languages.

Definition 2. *Let L be a language over vocabulary $V = \{v_1, \dots\}$, and $P = (V, C, \pi : V \mapsto C)$ a partition of V into categories C. Let $\pi(L)$ denote the language that is created by replacing a sequence of elements in V by the sequence of their categories.*

One defines the partition $P_{tot} = \{\{v\}, v \in V\}$ (one category per word) and the partition $P_{nul} = \{V\}$ (every word in the same category).

P_{tot} is such that $\pi_{tot}(L) \sim L$. The minimal partition P_{nul} does not contain any information.

A partition $P = (V, C, \pi)$ is contextual if it is impossible to determine word order in language L from sequences of its categories:

Definition 3. *Let L be a language over vocabulary V, and let $P = (V, C, \pi)$ be a partition over V. The partition P is said to be* contextual *if*

$$\pi(L) = \pi(\overline{L})$$

The trivial partition P_{nul} is always contextual.

Example. *Consider the natural language sample. We refer to the words by their initial letters: r(ats),e(at)..., thus we have $V = \{c, e, r, f, m, p, t, d, h, w\}$. and $L = \{cer, rfc, mpt, dhw\}$.*

One can check that the partition P_1 :

$$c_1 = \{c, r, e, f\}$$

$$c_2 = \{m, p, t\}$$

$$c_3 = \{d, h, w\}$$

is contextual: the well-formed sequences over this partition are $c_1c_1c_1$, $c_2c_2c_2$ and $c_3c_3c_3$. These patterns convey the information that words like 'mathematicians' and 'theorems' occur together, but do not provide information on order. Therefore $\pi_1(L) = \{c_1c_1c_1, c_2c_2c_2, c_3c_3c_3\} = \pi_1(\overline{L})$. P_1 is also a maximal partition for that property: any further splitting leads to order-specific patterns. Intuitively, this partition corresponds to the semantic categories $Animals = \{r, c, e, f\}$, $Science = \{m, p, t\}$, and $Medicine = \{d, h, w\}$.

A syntactic partition has two characteristics: its patterns encode the structure (in our case, order), and it is completely autonomous with respect to contextual information. Let us now express this autonomy formally.

Two partitions of the same vocabulary are said to be independent if they do not share any information with respect to language L. In other words, if we translate a sequence of symbols from L into their categories from one partition, this sequence of categories will not provide any information on how the sequence translates into categories from the other partition:

Definition 4. *Let L be a language over vocabulary V, and let $P = (V, C, \pi)$ and $P' = (V, C', \pi')$ be two partitions of V. P and P' are considered as independent with respect to L if*

$$\forall c_{i_1} \dots c_{i_n} \in \pi(L), \forall c'_{j_1} \dots c'_{j_n} \in \pi'(L)$$

$$\pi^{-1}(c_{i_1} \dots c_{i_n}) \cap \pi'^{-1}(c'_{j_1} \dots c'_{j_n}) \neq \emptyset$$

Definition 5. *Let L be a language over V, and let $P = (V, C, \pi)$ be a partition. P is said to be* syntactic *if it is independent of any contextual partition of V.*

A syntactic partition is hence a partition that does not share any information with contextual partitions; or, in linguistic terms, a syntactic pattern is equally applicable to any contextual category.

Example. *We can see that the partition P_2 :*

$$c_4 = \{c, r, m, t, d, w\}$$

$$c_5 = \{e, f, p, h\}$$

is independent of the partition P_1: one has $\pi_2(L) = \{c_4c_5c_4\}$. Knowing the sequence $c_4c_5c_4$ does not provide any information on which P_1 categories the words belong to. P_2 is therefore a syntactic partition.

Looking at the corpus, one might be tempted to consider a partition P_3 that sub-divides c_4 into subject nouns, object nouns, and - if one word can be mapped to only one category - "ambiguous" nouns:

$$c_6 = \{m, d\}$$

$$c_7 = \{t, w\}$$

$$c_8 = \{c, r\}$$

$$c_9 = \{e, f, p, h\}$$

The patterns corresponding to this partition would be $\pi_3(L) = \{c_6c_9c_7, c_8c_9c_8\}$. These patterns will not predict that sentence (2) is grammatical, because the word *wounds* was only seen as an object. If we want to learn the correct generalization we need to reject this partition in favour of P_2.
This is indeed what happens by virtue of definition 5. We notice that the patterns over P_3 categories are not independent of the contextual partition P_1: one can deduce from the rule $c_8c_9c_8$ that the corresponding sentence cannot be e.g. category c_2:

$$\pi_3^{-1}(c_8c_9c_8) \cap \pi_1^{-1}(c_2c_2c_2) = \emptyset$$

P_3 is hence rejected as a syntactic partition.

P_2 is the maximal syntactic partition: any further distinction that does not conflate P_1 categories would lead to an inclusion of contextual information. We can indeed see that category c_4 corresponds to *Noun* and c_5 corresponds to *Verb*. The syntactic rule for the sample is *Noun Verb Noun*. It becomes possible to distinguish between syntactic and contextual acceptability: *cats rats fear* is acceptable as a contextual pattern $c_1c_1c_1$ under *'Animals'*, but not a valid syntactic pattern. The sequence *wounds eat theorems* is syntactically well-formed by $c_5c_6c_5$, but does not correspond to a valid contextual pattern.

In this section we provided the formal definitions of syntactic information and the broader contextual information. By an illustrative example we gave an intuition of how we apply the autonomy of syntax principle in a non probabilistic grammar. We now turn to the probabilistic scenario and the inference from a corpus.

4 Syntactic and contextual categories in a corpus

As we have seen in section 2, probabilistic language modeling with a likelihood maximization objective does not have incentive to concentrate on syntactic generalizations. In what follows, we demonstrate that using the autonomy of syntax principle it is possible to infer syntactic categories for a probabilistic language.

A stochastic language L is a language which assigns a probability to each sequence. As an illustration of such a language, we consider the empirical distribution induced from the sample in section 3.

$$L = \{cer(\frac{1}{4}), rfc(\frac{1}{4}), mpt(\frac{1}{4}), dhw(\frac{1}{4})\}$$

We will denote by $p_L(v_{i_1} \dots v_{i_n})$ the probability distribution associated to L.

Definition 6. *Let V be a vocabulary. A* (probabilistic) partition *of V is defined by $P = (V, C, \pi : V \mapsto \mathbb{P}(C))$ where $\mathbb{P}(C)$ is the set of probability distributions over C.*

Example. *The following probabilistic partitions correspond to the non-probabilistic partitions (contextual and syntactic, respectively) defined in section 3. We will now consider these partitions in the context of the probabilistic language L.*

$$\pi_1 = \begin{array}{cc} \begin{array}{c} c \\ r \\ e \\ f \\ m \\ p \\ t \\ d \\ h \\ w \end{array} & \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \end{array}, \pi_2 = \begin{array}{cc} \begin{array}{c} c \\ r \\ e \\ f \\ m \\ p \\ t \\ d \\ h \\ w \end{array} & \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \end{array}$$

From a probabilistic partition $P = (V, C, \pi)$ as defined above, one can map a stochastic language L to a stochastic language $\pi(L)$ over the sequences of categories:

$$p_\pi(c_{i_1} \dots c_{i_n}) =$$

$$\sum_{u_{j_1} \dots u_{j_n}} (\prod_k \pi(c_{i_k} | u_{j_k})) p_L(u_{j_1} \dots u_{j_n})$$

As in the non-probabilistic case, the language $\overline{L}$ will be defined as the language obtained by shuffling the sequences in L.

Definition 7. *Let L be a stochastic language over vocabulary V. We will denote by $\overline{L}$ the language obtained by shuffling the elements in the sequences* *of L in the following way: for a sequence $v_1 \dots v_n$, one has*

$$p_{\overline{L}}(v_1 \dots v_n) = \frac{1}{n!} \sum_{(i_1 \dots i_n) \in \sigma(n)} p_L(v_{i_1} \dots v_{i_n})$$

One can easily check that $\pi(\overline{L}) = \overline{\pi(L)}$.

Example. *The stochastic patterns of L over the two partitions are, respectively:*

$$\pi_1(L) = \{c_1c_1c_1(\frac{1}{2}), c_2c_2c_2(\frac{1}{4}), c_3c_3c_3(\frac{1}{4})\}$$
$$\pi_2(L) = \{c_4c_5c_4(1)\}$$

We can now define a probabilistic contextual partition:

Definition 8. *Let L be a stochastic language over vocabulary V, and let $P = (V, C, \pi)$ be a probabilistic partition. P will be considered as* contextual *if*

$$\pi(L) = \pi(\overline{L})$$

We now want to express the independence of syntactic partitions from contextual partitions. The independence of two probabilistic partitions can be construed as an independence between two random variables:

Definition 9. *Consider two probabilistic partitions $P = (V, C, \pi)$ and $P' = (V, C', \pi')$. We will use the notation*

$$(\pi \cdot \pi')_v(c_i, c'_j) = \pi_v(c_i)\pi'_v(c'_j)$$

and the notation

$$P \cdot P' = (V, C \times C', \pi \cdot \pi')$$

P and P' are said to be independent (with respect to L) if the distributions inferred over sequences of their categories are independent:

$$\forall w \in \pi(L), \forall w' \in \pi'(L),$$
$$p_{\pi \cdot \pi'}(w, w') = p_\pi(w) p_{\pi'}(w')$$

A syntactic partition will be defined by its independence from contextual information:

Definition 10. *Let P be a probabilistic partition, and L a stochastic language. The partition P is said to be* syntactic *if it is independent (with respect to L) of any possible probabilistic contextual partition in L.*

Example. *The partition P_1 is contextual, as $\pi_1(L) = \overline{\pi_1(L)}$. The partition P_2 is clearly independent of P_1 w.r.t. L.*

4.1 Information-theoretic formulation

The definitions above may need to be relaxed if we want to infer syntax from natural language corpora, where strict independence cannot be expected. We propose to reformulate the definitions of contextual and syntactic information in the information theory framework.

We present a relaxation of our definition based on Shannon's information theory (Shannon, 1948). We seek to quantify the amount of information in a partition $P = (V, C, \pi)$ with respect to a language L. Shannon's entropy provides an appropriate measure. Applied to $\pi(L)$, it gives

$$H(\pi(L)) = -\sum_{w \in \pi(L)} p_\pi(w)(\log(p_\pi(w)))$$

For a simpler illustration, from now on we will consider only languages composed of fixed-length sequences s, i.e $|s| = n$ for a given n. If L is such a language, we will consider the language $\overline{\overline{L}}$ as the language of sequences of size n defined by

$$p_{\overline{\overline{L}}}(v_{i_1} \dots v_{i_n}) = \prod_j p_L(v_{i_j})$$

where $p_L(v)$ is the frequency of v in language L.

Proposition 1. *Let L be a stochastic language, $P = (V, C, \pi)$ a partition. One has:*

$$H(\pi(\overline{\overline{L}})) \geq H(\pi(\overline{L})) \geq H(\pi(L))$$

with equality iff the stochastic languages are equal.

Let C be a set of categories. For a given distribution over the categories $p(c_i)$, the partition defined by $\pi(c_i|v) = p(c_i)$ (constant distribution w.r.t. the vocabulary) contains no information on the language. One has $p_\pi(c_{i_1} \dots c_{i_k}) = p(c_{i_1}) \dots p(c_{i_k})$, which is the unigram distribution, in other words $\pi(L) = \pi(\overline{\overline{L}})$. As the amount of syntactic or contextual information contained in $\overline{\overline{L}}$ can be considered as zero, a consistent definition of the information would be:

Definition 11. *Let $P = (V, C, \pi)$ be a partition, and L a language. The information contained in P with respect to L is defined as*

$$I_L(P) = H(\pi(\overline{\overline{L}})) - H(\pi(L))$$

Lemma 1. *Information $I_L(P)$ defined as above is always positive. One has $I_{\overline{L}}(P) \leq I_L(P)$, with equality iff $\pi(\overline{L}) = \pi(L)$.*

After having defined how to measure the amount of information in a partition with respect to a language, we now translate the independence between two partitions into the terms of mutual information:

Definition 12. *We follow notations from Definition 9. We define the mutual information of two partitions $P = (V, C, \pi)$ et $P' = (V, C', \pi')$ with respect to L as*

$$I_L(P; P') = H(P) + H(P') - H(P \cdot P')$$

This directly implies that

Lemma 2. *$P = (V, C, \pi)$ and $P' = (V, C', \pi')$ are independent w.r.t. L*

$$\Leftrightarrow I_L(P; P') = 0$$

Proof. This comes from the fact that, by construction, the marginal distributions of $\pi \cdot \pi'$ are the distributions π and π'. □

With these two definitions, we can now propose an information-theoretic reformulation of what constitutes a contextual and a syntactic partition:

Proposition 2. *Let L be a stochastic language over vocabulary V, and let $P = (V, C, \pi)$ be a probabilistic partition.*

- *P is* contextual *iff*

$$I_L(P) = I_{\overline{L}}(P)$$

- *P is* syntactic *iff for any contextual partition P_**

$$I_L(P; P_*) = 0$$

4.2 Relaxed formulation

If we deal with non artificial samples of natural language data, we need to prepare for sampling issues and word (form) ambiguities that make the above formulation of independence too strict. Consider for instance adding the following sentence to the previous sample:

doctors heal fear

The distinction between syntactic and contextual categories is not as clear as before. We need a relaxed formulation for real corpora: we introduce γ-contextual and μ, γ-syntactic partitions.

Definition 13. *Let L be a stochastic language.*

- *A partition P is considered as* γ-contextual *if it minimizes*

$$I_L(P)(1 - \gamma) - I_{\overline{L}}(P) \qquad (1)$$

- *A partition P is considered μ, γ*-syntactic *if it minimizes*

$$\max_{P^*} I_L(P; P_*) - \mu\, I_L(P) \qquad (2)$$

for any γ-contextual partition P^*.

Let P and P' be two partitions for L, such that

$$\Delta_I(L) = I_{P'}(L) - I_P(L) \geq 0$$

then the γ-contextual program (1) would choose P' over P iff

$$\frac{\Delta_I(L) - \Delta_I(\overline{L})}{\Delta_I(L)} \leq \gamma$$

Let P^* be a γ-contextual partition. Let

$$\Delta_{MI}(L, P^*) = I_L(P'; P^*) - I_L(P; P^*)$$

then the μ, γ-syntactic program (2) would choose P' over P iff

$$\frac{\Delta_{MI}(L, P^*)}{\Delta_I(L)} \leq \mu$$

Example. *Let us consider the following partitions:*
- P_1 and P_2 refer to the previous partitions above: {Animals, Science, Medicine} and {Noun, Verb}
- P_A is adapted from P_1 so that 'fear' belongs to Animals and Medicine

$$\{c, e, r, f(\tfrac{1}{2})\}, \{m, p, t\}, \{d, h, w, f(\tfrac{1}{2})\}$$

- P_B merges Animals and Medicine from P_1

$$\{c, e, r, f, d, h, w\}, \{m, p, t\}$$

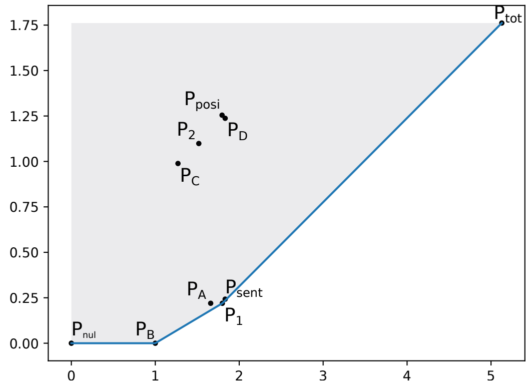

Figure 1: $I_L(P) - I_{\overline{L}}(P)$ represented w.r.t. $I_L(P)$ for different partitions: acceptable solutions of program (1) lie on the convex hull boundary of the set of all partitions. Solution for γ is given by the tangent of slope γ. Non trivial solutions are P_B and P_1.

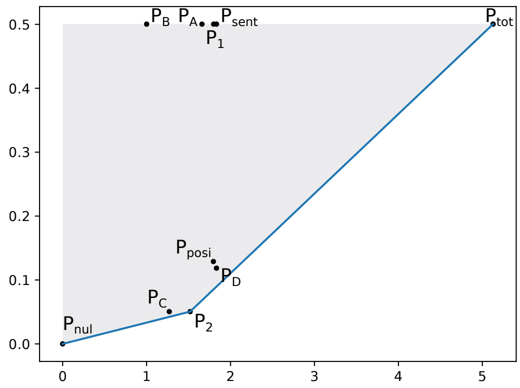

Figure 2: $I_L(P;P_B)$ represented w.r.t. $I_L(P)$ for different partitions: acceptable solutions of program (2) lies on the convex hull boundary of the set of all partitions. Solution for μ is given by the tangent of slope μ. Non-trivial solution is P_2.

- P_{sent} describes the probability for a word to belong to a given sentence (5 categories)
- P_C is adapted from P_2 so that 'fear' belongs to Verb and Noun

$$\{c, r, m, t, d, w, f(\tfrac{1}{2})\}, \{e, p, h, f(\tfrac{1}{2})\}$$

- P_D is adapted from P_2 and creates a special category for 'fear'

$$\{c, r, m, t, d, w\}, \{e, p, h\}, \{f\}$$

- P_{posi} describes the probability for a word to appear in a given position (3 categories)

Acceptable solutions of (1) and (2) are, respectively, on the convex hull boundary in Fig.1 and Fig.2. While the lowest parameter (non trivial) solutions are P_B for context and P_2 for syntax, one can check that partitions P_1, P_A and P_{sent} are all close to the boundary in Fig.1, and that partitions P_C, P_D and P_{posi} are all close to the boundary in Fig.2, as expected considering their information content.

4.3 Experiments

In this section we illustrate the emergence of syntactic information via the application of objectives (1) and (2) to a natural language corpus. We show that the information we acquire indeed translates into known syntactic and contextual categories.

For this experiment we created a corpus from the Simple English Wikipedia dataset (Kauchak, 2013), selected along three main topics: *Numbers*, *Democracy*, and *Hurricane*, with about 430 sentences for each topic and a vocabulary of 2963 unique words. The stochastic language is the set L^3 of 3-gram frequencies from the dataset. In order to avoid biases with respect to the final punctuation, we considered overlapping 3-grams over sentences. For the sake of evaluation, we construct one contextual and one syntactic embedding for each word. These are the probabilistic partitions over gold standard contextual and syntactic categories. The contextual embedding P_{con} is defined by relative frequency in the three topics. The results for this partition are $I_{L^3}(P_{con}) = 0.06111$ and $I_{\overline{L^3}}(P_{con}) = 0.06108$, corresponding to a γ threshold of $6.22.10^{-4}$ in (1), and thus distribution over topics can be considered as an almost purely contextual partition. The syntactic partition P_{syn} is the distribution over POS categories (tagged with the Stanford tagger, Toutanova et al. (2003)).

Using the gold categories, we can manipulate the information in the partitions by merging and splitting across contextual or syntactic categories. We study how the information calculated by (1) and (2) evolve; we validate our claims if we can deduce the nature of information from these statistics.

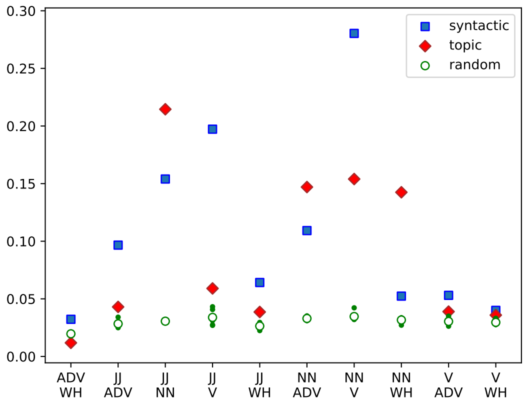

Figure 3: Increase of information Δ_I in three scenarios: syntactic split, topic split and random split.

We start from the syntactic embeddings and we split and merge over the following POS categories: Nouns (*NN*), Adjectives (*JJ*), Verbs (*V*), Adverbs(*ADV*) and Wh-words (*WH*). For a pair of categories (say *NN*+*V*), we create:

- P_{merge} merges the two categories $(NN + V)$
- P_{syntax} splits the merged category into NN and V (syntactic split)
- P_{topic} splits the merged category into $(NN + V)_{t_1}$, $(NN + V)_{t_2}$ and $(NN + V)_{t_3}$ along the three topics (topic split)

- P_{random} which splits the merged category into $(NN+V)_1$ and $(NN+V)_2$ randomly (random split)

It is clear that each split will increase the information compared to P_{merge}. We display the simple information gains Δ_I in Fig.3. The question is whether we can identify if the added information is syntactic or contextual in nature, i.e. if we can find a μ for which the μ, γ-syntactic program (2) selects every syntactic splitting and rejects every contextual or random one.

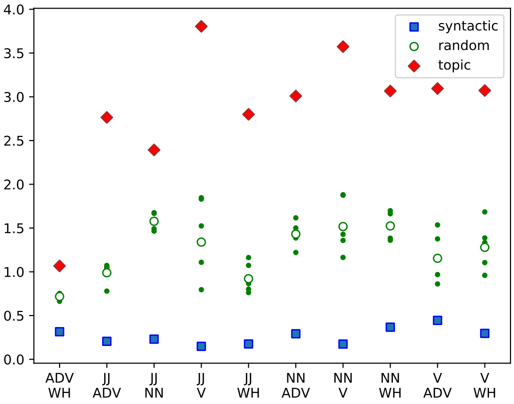

Figure 4: Ratio Δ_{MI}/Δ_I in three scenarios: syntactic split, topic split and random split. Considering objective (2) with parameter $\mu = 0.5$ leads to discrimination between contextual and syntactic information.

Fig.4 represents the ratio between the increase of mutual information (relatively to P_{con}) Δ_{MI} and the increase of information Δ_I, corresponding to the the threshold μ in (2). It shows that indeed for a $\mu = 0.5$ syntactic information (meaningful refinement according to POS) will be systematically selected, while random or topic splittings will not. We conclude that even for a small natural language sample, syntactic categories can be identified based on statistical considerations, where a language model learning algorithm would need further information or hypotheses.

4.4 Integration with Models

We have shown that our framework allows to search for syntactic categories without prior hypothesis of a particular model. Yet if we do have a hypothesis, we can indeed search for the syntactic categories that fit the particular class of models $\mathcal{M}$. In order to find the categories which correspond to the syntax rules that can be formulated in a given class of models, we can integrate the model class in the training objective by replacing entropy by the negative log-likelihood of the training sample.

Let $M \in \mathcal{M}$ be a model, which takes a probabilistic partition $P = (V, C, \pi)$ as input, and let $LL(M, P, L_S)$ be the log-likelihood obtained for sample S. We will denote

$$\tilde{H}(L_S, P) = -\sup_{M \in \mathcal{M}} LL(M, P, L_S)$$

$$\tilde{I}_{L_S}(P) = \tilde{H}(\overline{\overline{L_S}}, P) - \tilde{H}(L_S, P)$$

Following Definition 12, we define

$$\tilde{I}_{L_S}(P; P') = \tilde{H}(L_S, P) + \tilde{H}(L_S, P') - \tilde{H}(L_S, P \cdot P')$$

We may consider the following program:

- A partition P is said to be γ-*contextual* if it minimizes
$$\tilde{I}_{L_S}(P)(1-\gamma) - \tilde{I}_{\overline{L_S}}(P)$$
- Let P_* be a γ-contextual partition for L, $\mu \in \mathbb{R}^+$, $k \in \mathbb{N}$. The partition P is considered μ, γ-syntactic if it minimizes
$$\max_{P^*} \tilde{I}_{L_S}(P; P^*) - \mu\, \tilde{I}_{L_S}(P)$$

5 Conclusion and Future Work

In this paper, we proposed a theoretical reformulation for the problem of learning syntactic information from a corpus. Current language models have difficulty acquiring syntactically relevant generalizations for diverse reasons. On the one hand, we observe a natural tendency to lean towards shallow contextual generalizations, likely due to the maximum likelihood training objective. On the other hand, a corpus is not representative of human linguistic competence but of performance. It is however possible for linguistic competence - syntax - to emerge from data if we prompt models to establish a distinction between syntactic and contextual (semantic/pragmatic) information.

Two orientations can be identified for future work. The immediate one is experimentation. The current formulation of our syntax learning scheme needs adjustments in order to be applicable to real natural language corpora. At present, we are working on an incremental construction of the space of categories.

The second direction is towards extending the approach to morphologically rich languages. In that case, two types of surface realization need to be considered: word order and morphological markers. An agglutinating morphology probably allows a more straightforward application of the method, by treating affixes as individual elements of the vocabulary. The adaptation to other types of morphological markers will necessitate more elaborate linguistic reflection.

References

Kuncoro Adhiguna, Dyer Chris, Hale John, Yogatama Dani, Clark Stephen, and Blunsom Phil. 2018. LSTMs can learn syntax-sensitive dependencies well, but modeling structure makes them better. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 1426–1436.

Marco Baroni. 2019. Linguistic generalization and compositionality in modern artificial neural networks. *CoRR*, abs/1904.00157.

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2017. What do neural machine translation models learn about morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 861–872.

Iris Berent and Gary Marcus. 2019. No integration without structured representations: Response to Pater. *Language*, 95:1:e75–e86.

Jean-Philippe Bernardy and Shalom Lappin. 2017. Using deep neural networks on learn syntactic agreement. *Linguistic Issues in Language Technology*, 15(2):1–15.

Terra Blevins, Omer Levy, and Luke Zettlemoyer. 2018. Deep RNNs encode soft hierarchical syntax. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 14–19.

Samuel R. Bowman, Christopher D. Manning, and Christopher Potts. 2015. Tree-structured composition in neural networks without tree-structured architectures. In *NIPS Workshop on Cognitive Computation: Integrating Neural and Symbolic Approaches*.

Noam Chomsky. 1957. *Syntactic Structures*. Mouton, Berlin, Germany.

Noam Chomsky. 1980. Rules and representations. *Behavioral and Brain Sciences*, 3(1):1–15.

Shammur Absar Chowdhury and Roberto Zamparelli. 2018. RNN simulations of grammaticality judgments on long-distance dependencies. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 133–144.

Alexander Clark and Rémi Eyraud. 2006. Learning auxiliary fronting with grammatical inference. In *Conference on Computational Language Learning*, pages 125–132.

Alexander Clark and Shalom Lappin. 2010. Unsupervised learning and grammar induction. In *Handbook of Computational Linguistics and Natural Language Processing*. Wiley-Blackwell, Oxford.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 2126–2136.

Chris Dyer, Adhiguna Kuncoro, Miguel Ballesteros, and Noah A. Smith. 2016. Recurrent neural network grammars. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Gottlob Frege. 1892. Über Sinn und Bedeitung. *Zeitschrift für Philosophie und philosophische Kritik*, 100:25–50.

Mario Giulianelli, Jack Harding, Florian Mohnert, Dieuwke Hupkes, and Willem Zuidema. 2018. Under the hood: Using diagnostic classifiers to investigate and improve how language models track agreement information. In *EMNLP Workshop Blackbox NLP: Analyzing and Interpreting Neural Networks for NLP*, pages 240–248.

E. Mark Gold. 1967. Language identification in the limit. *Information and control*, 10:5:447–474.

Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. 2018. Colorless green recurrent networks dream hierarchically. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1195–1205.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the North American Chapter of the Association for Computational Linguistics*.

Dieuwke Hupkes, Sara Veldhoen, and Willem H. Zuidema. 2018. Visualisation and 'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure. *Journal of Artificial Intelligence Research*, 61:907–926.

David Kauchak. 2013. Improving text simplification language modeling using unsimplified text data. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 1537–1546.

Brenden M. Lake and Marco Baroni. 2017. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *34th International Conference on Machine Learning*.

Shalom Lappin and Stuart Shieber. 2007. Machine learning theory and practice as a source of insight into universal grammar. *Journal of Linguistics*, 43:393–427.

Beth Levin and Malka Rappaport Hovav. 2005. *Argument Realization*. Cambridge University Press, Cambridge.

Omer Levy, Steffen Remus, Chris Biemann, and Ido Dagan. 2015. Do supervised distributional methods really learn lexical inference relations? In *Proceedings of the North American Chapter of the Association for Computational Linguistics Human Language Technologies*, pages 970–976, Denver, Colorado. Association for Computational Linguistics.

Xiang Lisa Li and Jason Eisner. 2019. Specializing word embeddings (for parsing) by information bottleneck. In *2019 Conference on Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing*, pages 2744–2754, Hong Kong, China.

Tal Linzen. 2019. What can linguistics and deep learning contribute to each other? Response to Pater. *Language*, 95(1):e98–e108.

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.

Rebecca Marvin and Tal Linzen. 2018. Targeted syntactic evaluation of language models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium. Association for Computational Linguistics.

Richard McCoy, Robert Frank, and Tal Linzen. 2018. Revisiting the poverty of the stimulus: hierarchical generalization without a hierarchical bias in recurrent neural networks. *ArXiv*, abs/1802.09091.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448.

Tomas Mikolov, Martin Karafiat, Lukas Burget, Jan Cernocky, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *INTERSPEECH*.

Frederick J. Newmeyer. 2003. Grammar is grammar and usage is usage. *Language*, 79:4:682–707.

Timothy Niven and Hung-Yu Kao. 2019. Probing neural network comprehension of natural language arguments. In *Proceedings of the 57th Annual Meeting of the Association for Computa-tional Linguistics*, pages 4658–4664.

Joe Pater. 2019. Generative linguistics and neural networks at 60: Foundation, friction, and fusion. *Language*, 95:1:41–74.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018a. Deep contextualized word representations. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Matthew E. Peters, Mark Neumann, Luke Zettlemoyer, and Wentau Yih. 2018b. Dissecting contextual word embeddings: Architecture and representation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509.

Shauli Ravfogel, Yoav Goldberg, and Tal Linzen. 2019. Studying the inductive biases of RNNs with synthetic variations of natural languages. *CoRR*, abs/1903.06400.

Shauli Ravfogel, Yoav Goldberg, and Francis Tyers. 2018. Can LSTM learn to capture agreement? the case of basque. In *EMNLP Workshop Blackbox NLP: Analyzing and Interpreting Neural Networks for NLP*, pages 98–107, Brussels, Belgium.

Naomi Saphra and Adam Lopez. 2018. Language models learn POS first. In *EMNLP Workshop Blackbox NLP: Analyzing and Interpreting Neural Networks for NLP*, pages 328–330, Brussels, Belgium. Association for Computational Linguistics.

David Saxton, Edward Grefenstette, Felix Hill, and Pushmeet Kohli. 2019. Analysing mathematical reasoning abilities of neural models. In *Proceedings of the 7th International Conference on Learning Representations*.

Marten van Schijndel, Aaron Mueller, and Tal Linzen. 2019. Quantity doesn't buy quality syntax with neural language models. In *Proceedings of Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing*, pages 5830–5836. Association for Computational Linguistics.

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

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Sam Bowman, Dipanjan Das, and Ellie Pavlick. 2019. What do you learn from context? Probing for sentence structure in contextualized word representations. In *International Conference on Learning Representations*.

Naftali Tishby, Fernando Pereira, and William Bialek. 1999. The information bottleneck method. In *Annual Allerton Conference on Communication, Control and Computing*, pages 368–377.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the North American Chapter of the Association for Computational Linguistics*, page 173180.