Modeling word interpretation with deep language models: The interaction between expectations and lexical information

Laura Aina (laura.aina@upf.edu)
Universitat Pompeu Fabra, Barcelona, Spain

Thomas Brochhagen (thomasbrochhagen@gmail.com)
Universitat Pompeu Fabra, Barcelona, Spain

Gemma Boleda (gemma.boleda@upf.edu)
Universitat Pompeu Fabra, Barcelona, Spain / ICREA, Barcelona, Spain

Abstract

How a word is interpreted depends on the context it appears in. We study word interpretation leveraging deep language models, tracing the contribution and interaction of two sources of information that have been shown to be central to it: context-invariant lexical information, represented by the word embeddings of a model, and a listener's contextual expectations, represented by its predictions. We define operations to combine these components to obtain representations of word interpretations. We instantiate our framework using two English language models, and evaluate the resulting representations in the extent by which they reflect contextual word substitutes provided by human subjects. Our results suggest that both lexical information and expectations codify information pivotal to word interpretation; however, their combination is better than either on its own. Moreover, the division of labor between expectations and the lexicon appears to change across contexts.

Keywords: expectations; word meaning; language models; distributional semantics; deep learning; ambiguity

Introduction

The ability to convey a potentially unbounded number of meanings is a central property of natural language. One prominent way to convey meaning is through composition, by forming composite meaning based on the meaning of constituents and their syntactic combination (Partee, 1995). For instance, the interpretation of *red box* is contingent on the meaning of *red* and on that of *box*. However, the semantic contribution of constituents themselves can also vary as a function of the context they appear in: the hue of *red* can differ when combined with either *wine*, *light*, or *blood*; the subject in *the mouse fled* is likely interpreted differently in *the mouse broke*; and *show* fulfills a different function in *I show a picture* than in *I went to the show*. Whether in terms of discrete or nuanced distinctions among senses, lexical ambiguity is pervasive in language (Cruse, 1986).

The way context affects a word's interpretation has been characterized in many ways, depending –among others and non-exclusively– on whether lexical meaning is taken to be rich in information (e.g., Pustejovsky, 1995) or rather underspecified (e.g., Frisson, 2009); on whether it is taken to be retrieved (e.g., Foraker & Murphy, 2012), modulated (e.g., Recanati, 2004) or instead constructed on the fly (e.g., Casasanto & Lupyan, 2015); and on whether the underlying process is taken to be driven by prediction (e.g., Kutas, DeLong, & Smith, 2011; Pickering & Garrod, 2007), relevance (e.g., Wilson & Carston, 2007), or reasoning about language use (e.g., Kao, Bergen, & Goodman, 2014). These differences notwithstanding, most analyses of contextualized word meaning share two assumptions (see Rodd 2020 for a review): (1) Some context-invariant lexical information comes into play when interpreting words; and (2) contextual expectations influence interpretation through an anticipation of what the interlocutor will (intend to) convey.

Taking up these leads, we use computational data-driven methods to explore how much headway into contextualized word meaning can be made by a concrete instantiation of these assumptions. To this end, we leverage two components of state-of-the-art language models: deep learning models trained to predict a word given its linguistic context. On the one hand, we model word expectancy in context through language models' probabilistic output (Armeni, Willems, & Frank, 2017). Intuitively, this component stands in for a listener's expectations (over words) given the context of utterance. On the other hand, we represent lexical information via the distributed word representations of the model, as an instance of Distributional Semantics (Boleda, 2020). Intuitively, this is a listener's lexicon: word-related knowledge abstracted over many instances of language use. We obtain representations of expectations and lexical information in the same multi-dimensional space from these two components. We then define operations over them to obtain representations of contextual word meaning, requiring neither ad-hoc training nor top-down specifications (e.g., of semantic features). Figure 1 sketches out our models' architecture.

This operationalization of the interaction between expected and lexical information bears strong ties to (1) pragmatic accounts in which a listener's interpretation is contingent on expectations about a speaker's language use, such as the Rational Speech Acts model (Goodman & Frank, 2016) and Relevance Theory (Wilson & Carston, 2007), and (2) processing accounts that factor in word expectancy (Surprisal Theory; Levy, 2008); for instance, due to the need of adjusting pre-activated expected information to the actual input (Predictive Coding; Huang & Rao, 2011).

In order to elucidate whether and to which degree our computational models of expectations; or of the lexicon; or of their combination can capture word interpretation, we evaluate them on a large-scale word substitution task (Kremer, Erk, Padó, & Thater, 2014). Our analysis shows that the combination of both information sources is best at representing con-

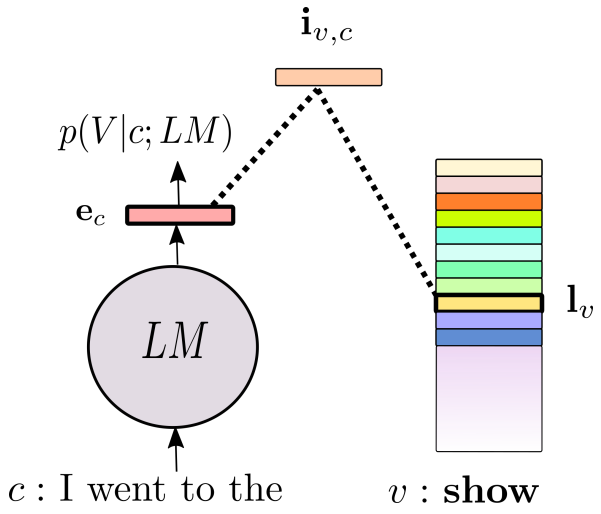

Figure 1: The context of utterance c is processed by the language model to yield an expectation $\mathbf{e}_c$, which defines a distribution over the vocabulary given the context, $p(V \mid c;LM)$. The context-invariant word embedding of a word v, $\mathbf{l}_v$, represents lexical information. Both information sources are combined to yield a word's contextualized interpretation $\mathbf{i}_{v,c}$.

textual meaning. This suggests that these components capture complementary information. However, the optimal degree to which weight is put on one component over the other changes across cases. This, in turn, hints at a flexible division of labor between expectations and the lexicon, raising the question of which factors drive this variation.

Word intepretation: Context & Lexicon

Contextual expectations. That listeners' contextual expectations come into play when interpreting a word is clearly illustrated by garden path sentences such as *the old man the boat*, where a word's likely disambiguation given its preceding context – here, *old* as an adjective and *man* as a noun – can lead to a conflicting parse of the sentence (e.g., Fine, Jaeger, Farmer, & Qian, 2013). Expectation-driven effects can also be more subtle. For instance, *N*400 effects, commonly associated with interpretability issues, can be caused by semantically unexpected items (e.g., the object in *the cat picked up the chainsaw*; see Cosentino, Baggio, Kontinen, and Werning 2017; Filik and Leuthold 2008; Nieuwland and Berkum 2006, a.o.). Crucially, such effects disappear when sentences are embedded in supportive contexts (e.g., a story about an anthropomorphized lumberjack cat). Overall, this body of work suggests that contextual expectations are facilitatory for processing, appearing to guide interpretation. This idea is also found in pragmatic theories. Expectations of rational language use are central in the Gricean tradition (Goodman & Frank, 2016; Grice, 1975), with listeners and speakers reasoning about each other's language use. Within Relevance Theory (e.g. Wilson & Carston, 2007) a listener is taken to infer occasion-specific meanings based on the stand-alone concept encoded by a word and expectations about what is relevant for the speaker to convey.

We here use language models (LMs) to represent contextual listener expectations. LMs are trained on unstructured text corpora to predict a word given a linguistic context. As a result, they form expectations in terms of distributions over a vocabulary conditioned on a context (Figure 1). LMs have been used to derive estimates of word expectancy (or surprisal), shown to correlate with measures of cognitive cost like reading times (Smith & Levy, 2013) and *N*400 amplitude (Frank, Otten, Galli, & Vigliocco, 2013). In this study, we employ state-of-the-art neural network models. Their probabilistic output has been shown to be sensitive to various aspects of linguistic knowledge (such as syntax; Linzen, Dupoux, and Goldberg 2016 and subsequent work). Moreover, internal representations of these models have been employed as contextualized word representations (Peters et al., 2018). In this work, we also make use of internal representations from LMs. However, since we use them to represent expectations, we do not focus on states that reflect how the model processed a word – derived when it is inputted – but rather on those that are involved in its predictions (*current* vs. *predictive states* in Aina, Gulordava, and Boleda 2019).

Lexical information. By the *lexical information* of a word, we mean stored context-invariant information that provides a basis to infer its meanings in context. As mentioned above, while lexical information is commonly assumed to play a role in language interpretation (Rodd, 2020), there is little consensus about its precise nature (Falkum & Vicente, 2015): for instance, on whether each word sense is separately stored in the mental lexicon, or how sense frequency affects a lexical entry and its access.

We model lexical information using the distributed word representations, or *embeddings*, in a LM. Word embeddings can be seen as the module of a LM that functions as the lexicon. They are learned as a byproduct of the word prediction task: the more similar the distributional patterns of two words, the more similar their representations are optimized to be. Hence, they are one of many possible instantiations of distributional semantic models of word meaning (Boleda, 2020), equivalent for our purposes to other distributional models of the lexicon such as Latent Semantic Analysis (Landauer & Dumais, 1997). Our choice to represent lexical information through word embeddings is mainly motivated by (1) their empirical success at capturing lexical information in a variety of tasks, amply shown in computational linguistics (e.g., Mikolov, Yih, & Zweig, 2013), and (2) the fact that this allows us to represent lexical and contextual information in a compatible manner, such that we can examine the contribution of each component both in isolation and combination. In other words, our use of LMs' word embeddings to represent lexical information is a modeling choice. It does not necessarily imply our endorsement of Distributional Semantics as a cognitive model of the lexicon (see Günther, Rinaldi, and

Marelli 2019 for discussion).

Word embeddings, once learned, are static across contexts, and thus conflate all uses of a word form into a single rich representation (Camacho-Collados & Pilehvar, 2018). Several methods to contextualize the information in distributional representations have been proposed (Erk & Padó, 2008, a.o.,). Following this lead, we consider vector operations that combine the word embeddings with contextual information (here: a LM's expectations). Our proposal is close in spirit to the framework motivated by Rodd (2020). We share the interactive view put forward by many "constraint"-based models in assuming expectations to drive comprehension; and also work with distributed representations of lexical meaning. However, while Rodd envisions a one-to-many relation between forms and sense representations, we assign one representation to each word form and let senses emerge in context.

Computational models

As suggested by Figure 1, a neural network language model *LM* is trained to output a probability distribution over the vocabulary V given a context. Generically, $p(V \mid c) = LM(c)$. Each word is typically encoded as a vector – a word embedding – learned as part of training and static across contexts. These word embeddings are stored in an input matrix W_i ($n \times |V|$, where n is the size of the embedding). The vectors of the words in c are processed to yield intermediate representations within the hidden layers of *LM*, with details varying across architectures (e.g., RNN, Transformer). After processing c, given an activation vector $\mathbf{y}$ of size m, an output probability can be obtained by multiplying it with an output matrix W_o ($|V| \times m$), followed by softmax; $p(V \mid c) = softmax(W_o\mathbf{y})$.

In the following, we consider LM architectures where W_o is the matrix transpose of W_i (thus, $n = m$), meaning that the weights of the input and output matrix are shared (Inan, Khosravi, & Socher, 2017; Press & Wolf, 2017). This technique reduces the size of the model while often enhancing its quality. More central to our purposes, it enforces a correspondence between the input and the output space, which enables us to represent lexical information and expectations in a shared space.

Lexical information. We take a pre-trained LM's input matrix W_i to represent a lexicon: the lexical content of word v, $\mathbf{l}_v$, is then a row-vector of W_i. The similarity between word representations is reflected by the geometric proximity of their vectors (e.g., as measured by their cosine). Words that appear in similar contexts will have similar embeddings pointing – in a graded manner – toward shared features (e.g., semantic or morpho-syntactic). As mentioned earlier, such embeddings are abstraction over all uses of a word and thus may codify information relevant to different word senses. This information on its own – out of context – is unlikely be an adequate model of contextual word interpretation. Notwithstanding, it constrains the information that a word could potentially convey, and it subsumes the kind of ambiguity that listeners are faced with when encountering a word form in context.

Contextual expectations. To represent expectations so that we can combine them with lexical information, our desideratum is a multi-dimensional representation in the same space of word embeddings, such that its proximity relations to word embeddings reflect how expected, or predictable, a word is in a certain context; that is, its probability. We devise a method that satisfies these criteria, leveraging activations from a pre-trained *LM* with weight sharing between the input and output matrix. In this case, the multiplication between $\mathbf{y}$ and W_o, resulting in the output scores, is equivalent to the dot product between $\mathbf{y}$ and each word embedding in W_i – our lexicon. This implies that the output probabilities are dependent on the similarity – in terms of dot product – between $\mathbf{y}$ and the word embeddings (Gulordava, Aina, & Boleda, 2018). Thus, the position of $\mathbf{y}$ in the space is optimized to reflect the extent to which a word is expected by the *LM*, through the similarity to its embedding. For this reason, $\mathbf{y}$ meets our criteria for a vectorial representation of contextual expectations; we henceforth refer to it as $\mathbf{e}_c$. Its position in the word space encodes not only which words are highly expected, but also which features of them are: if an animate noun is expected, $\mathbf{e}_c$ will be closer to animate nouns. However, on its own, it may not be an adequate model of word interpretation. In forming expectations, the LM has not "seen" the target word: it consequently harbors uncertainty about its identity and meaning.

Operations. We define two operations that combine lexical information $\mathbf{l}_v$ about a word v with contextual expectations $\mathbf{e}_c$ into representations reflecting v's contextual interpretation: $\mathbf{i}_{v,c}$.[1] Both operations are parametrized by the degree to which they rely on either source of information:

(1) **Weighted average** (*avg*): $\mathbf{i}_{v,c} = (1-\alpha)\mathbf{e}_c + \alpha\,\mathbf{l}_v$

Intuitively, this operation blends expected and lexical information, highlighting what is common between the two: the part of the lexical information that is relevant to the context. Depending on $\alpha \in [0;1]$, the resulting vector is influenced more by one information source than the other: if $\alpha = 0$, contextualized interpretations are just expectations; conversely, if $\alpha = 1$, interpretation relies solely on lexical information.

(2) **Delta rule** (*delta*): $\mathbf{i}_{v,c} = \mathbf{e_c} - \alpha\nabla D_{\mathbf{e}_c}$, where $D = 1 - cos(\mathbf{e}_c, \mathbf{l}_v)$

This operation reduces the distance D between the vectors by shifting $\mathbf{e}_c$ in the direction of the negative gradient of D with respect to $\mathbf{e}_c$ ($\nabla D_{\mathbf{e}_c}$). α regulates how close expectations are pulled towards lexical information. If $\alpha = 0$, $\mathbf{i}_{v,c} = \mathbf{e}_c$. As α approaches infinity, the contribution of $\mathbf{l}_v$ grows and that of $\mathbf{e}_c$ shrinks. Intuitively, this operation is a form of "expectation revision", adapting formed expectations to the actual input.

[1] The vectors are normalized before these operations.

Our framework can be seen as modeling the way contextual expectations, as a result of top-down processing, are combined with word-level information, associated to the bottom-up input. Due to the way our model is designed, the geometric distance between the output of the interpretation process – $\mathbf{i}_{v,c}$ – and the expected information that was activated without seeing the word – $\mathbf{e}_c$ – is proportional to the surprisal relative to the word, as estimated by the LM.[2] The more surprising a word is, the further $\mathbf{i}_{v,c}$ is to $\mathbf{e}_c$: this can be seen as the extent that expectations need to be "changed" in order to accommodate the bottom-up input, which can be construed as a measure of cognitive cost associated with interpretation. Crucially, in our model, the magnitude of such distance is also modulated by the α parameter. This aspect of our model shows a strong tie with surprisal-based account of processing (Zarcone, Van Schijndel, Vogels, & Demberg, 2016), as well as to pragmatic accounts factoring in processing efforts in the comprehension of an utterance (Wilson & Sperber, 2006). In this study, we focus on evaluating the ability of our framework to account for the output on interpretation processes (i.e., how words are understood in context). However, we plan to investigate how our model relates to aspects of processing in future work.

Analysis

Data. To evaluate our models' components (lexical and expected information) and ways of combining them, we use Kremer et al.'s (2014) Concepts in Context (CoInCo) dataset, an English corpus annotated for *lexical substitution*. It contains 15.5K content words in context (at most 3 sentences), with at least 6 crowd-sourced paraphrases per word. In other words, this corpus lists alternatives that could appropriately substitute a target word in the context it appears in. They are proxies for word meaning in context and can be seen as the output of an offline interpretation task (Table 1 shows excerpts of CoInCo). This dataset enables us to test our framework: (1) on a large scale and a relatively natural distribution of words and meanings, and (2) on nuanced differences among word usages, without appeal to predefined lists of senses.

Models. We use two English language models to check the robustness of our methods and the trends we report. The first model is an LSTM, adapted from Aina et al. (2019). The second is the transformer-based BERT model in its *large* version (Devlin, Chang, Lee, & Toutanova, 2019). Both LMs employs the left and right context of a word when predicting it (*bidirectional*); have a weight tying mechanism; and $\mathbf{e}_c$ is the result of a non-linear transformation on the last hidden state.[3] The models are pre-trained and their weights are not updated during our evaluation. Besides the different architectures, these models differ in size and amount of training data. Since BERT drastically surpasses the LSTM with respect to both, we expect it to be better at word prediction, and consequently to have better representations of expectations. The choice of using bidirectional models is motivated by the data considered in this study, collected as an offline task where both the left and right context of a word were accessible. However, our framework can also be instantiated using unidirectional LMs, taking into account only one side of a context; this suggests potential links to incremental processing to be explored in future experiments.

Evaluation. Given a representation, we use cosine similarities between this and a word's embedding to estimate its plausibility as a substitute. Following previous research, we consider two tasks to assess the quality of a representation. The first task concerns ranking: we order all substitutes of a word type across the dataset by cosine to the evaluated vectors and compare the ranking to the datapoint's gold standard – GAP (Thater, Dinu, & Pinkal, 2009). The second task concerns retrieval: we take our 10 highest ranking word lemmas in terms of cosine and measure their overlap with substitutes provided by annotators – RECALL-10 (McCarthy & Navigli, 2007). Both GAP and RECALL-10 factor in the number of annotators that provided a particular substitute, with more weight put on those with higher agreement; for both scores, the higher the better. CoInCo comes with a *dev*/*test* split (10K/5K, filtered by coverage of the LMs).[4] We use *dev* data to inspect the influence of α in modulating between expectations and lexical information. For each LM and operation, we evaluate the data at 10 α-values, ranging over $[0,1]$ for *avg* and over $[0,3]$ for *delta*. We report: (1) mean performance over α-values with constant α across data points and (2) mean performance for optimal α per datum. We take an α-value to be optimal if it yields the highest sum of RECALL-10 and GAP score. Through (1), we identify the optimal constant α – the best across all data – for each combination of an LM with an operation. We then use this α for evaluation on *test* data. (2) is instead indicative of whether α better works as a datum-dependent or -independent parameter.

Results. Figure 2 shows how different ways of combining and weighting expectations and lexical information impact

[2] The distance between $\mathbf{i}_{v,c}$ and $\mathbf{e}_c$ depends, in the first place, on the distance between $\mathbf{e}_c$ and $\mathbf{l}_v$ (e.g., if they are close, their weighted average will remain close to them). As mentioned earlier, the dot product between these two is what the probability of v depends upon, and consequently its surprisal ($-logP(v|c;LM)$).

[3] Differently from Aina et al. (2019), we weight-tie the LSTM's input and output matrices and add a non-linear transformation between the last hidden layer and the output layer (BERT already comes with these features). This transformation reduces the conflation of expected words' representations and other information, such as the context (Gulordava et al., 2018). In the LSTM, the last hidden state is obtained processing the left and right context of a word; in BERT the whole text chunk is processed but with the target word masked. We refer to the aforementioned papers for details.

[4] We cover data points whose target word and at least one substitute are in our models' vocabularies. For BERT we also skip words split into subwords. This yields BERT: $\approx 90\%$; LSTM: $\approx 95\%$.

α	Abridged context (target word boldfaced)	Human substitutions	Model substitutions
0.0	(1) He accepts his role with a club and he's a team **guy**	player, man, dude, member, person	player, manager, man, captain, leader
0.5	(2) Some critics already say "Waking With Tasha" is Malaquez's finest **work**	masterpiece, enterprise, artwork, production, endeavor	performance, painting, production, project, job
1.0	(3) [He] had his own Jeep and went to the **beach**	shore, coast, dock, oceanfront, sand	shore, coast, waterfront, shoreline, coastline

Table 1: Abridged CoInCo datum with example LSTM outputs for weighted average α.

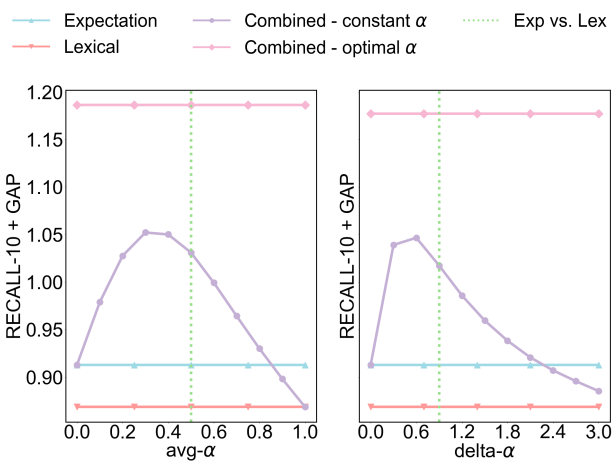

Figure 2: Results on *dev*-data for BERT. Dotted lines mark whether a given α yields a $\mathbf{i}_{v,c}$ that is closer to $\mathbf{e}_c$ (to the line's left) than to $\mathbf{l}_v$ (to its right).

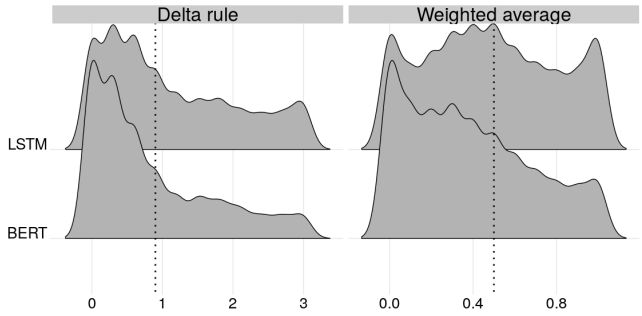

Figure 3: Distribution of optimal per-datum α. Dotted lines mark whether a given α yields a representation that is closer to $\mathbf{e}_c$ (to the line's left) than to $\mathbf{l}_v$ (to its right).

the performance on *dev* data, zooming in on BERT (similar trends are obtained for the LSTM). In a nutshell, relying solely on either expectations or lexical information is worse than combining them; this holds for both *avg* and *delta*. For the two operations, the best constant α values are intermediate ones where both sources of information play a role to some degree; performances degrade when using higher or lower values. With BERT, we observe a slight preference for expectations: the best constant α-values result in interpretations that are closer to $\mathbf{e}_c$ than to $\mathbf{l}_v$. Nevertheless, factoring in lexical information does substantially improve our contextualized representations. Lastly, note that the optimal α-values per datum far outperform any constant α, meaning that the balance between the optimal contribution of $\mathbf{e}_c$ and $\mathbf{l}_v$ varies across words and contexts. Figure 3 shows the distribution of optimal α. This result suggests that the extent expectations or the lexicon are to be trusted is a contextual matter. In future work, we plan to investigate which factors of a context and of a word drive this variation, and possibly extend our framework in order to dynamically modulate α.

Table 2 summarizes performance across models and operations on *test* data. On the one hand, since expectations harbor much uncertainty about the target, they often fail to have substitutes as the closest words (RECALL). However, when ranking word-specific candidates (GAP) their contextual responsiveness is more advantageous than pure lexical information about the target. As with *dev* data, the best contextual representations are obtained when lexical and expected information are combined, using either *avg* or *delta*. Both RECALL-10 and GAP increase, speaking to the relative appropriateness of the point in (vectorial) space in which our combined representations end up. In terms of overall results, *avg* and *delta* do not differ much in BERT, and are even identical in the LSTM. We leave the exploration of their differences for future investigation. More generally, the difference between the LSTM and BERT are largely attributable to the fact that BERT is a better LM and therefore has more sensible expectations, leading to higher reliance on them and better overall performance (Table 2). We hypothesize that, at least in this setup, the best constant α may decrease as a measure of the LM's quality (dependent on size; data; architecture), but also that it is unlikely to ever reach the value 0, where lexical information is ignored. It is clear from sentences like (3) in Table 1 that, while contexts provides hints and constraints with respect to what a word may convey (e.g., a location), knowing that *beach* was uttered and the information this word comes with is crucial for comprehension, due to the pervasive uncertainty that underlies in communication. Additionally, the setup of this study may impact what observed about the best constant α. First, providing both the left and right context of a word may largely reduce the uncertainty over expectations. Instantiating our framework with a left-to-right model may shed a different light on α. Second, as observed earlier,

	expected	lexical	avg	delta
	BERT (avg: α=0.3; delta: α=0.6)			
GAP	0.52	0.46	**0.54**	0.53
RECALL-10	0.34	0.43	0.49	**0.51**
	LSTM (avg: α=0.5; delta: α=0.9)			
GAP	0.45	0.45	**0.48**	**0.48**
RECALL-10	0.13	0.34	**0.38**	**0.38**

Table 2: *Test* results in ranking (GAP) and retrieving (RECALL-10) substitutes, with best constant α.

the extent that expectations are to be trusted for interpretation is better modeled as a contextual matter (Fig.2; constant vs. optimal α); a preference for expectations over lexical information does not hold for all situations.

Conclusion

Both lexical information and the expectations that a context gives rise to influence how words are interpreted. In this study, we proposed a computational model of word interpretation that makes use of deep language models' components: A listener's lexicon is represented using distributed word representations, and a listener's expectations as a function of a language model's probabilistic output. In our operationalization, the two draw from the same data-driven machinery – a language model – and operate in the same multi-dimensional representational space. An important benefit of our approach is that it can be flexibly applied to different combinations of words and contexts. This is possible due to (1) its reliance on pre-trained deep language models to obtain representations of expectations and the lexicon, and (2) the general method of obtaining interpretations through vector operations. Our framework resonates with various proposals on ambiguity resolution and processing, providing a methodology to instantiate and test some of their assumptions through computational data-driven modeling. We hope that the ideas we put forward will inspire new applications of deep language models in Linguistics and Cognitive Science.[5]

We instantiated our framework using two English language models, and used it to investigate the interaction between the lexicon and expectations in word interpretation, considering a large-scale lexical substitution task. Our results suggest that these two sources codify (at least partially) complementary information, with representations drawing from a combination of both performing best. More broadly, we find that the degree to which each source contributes to contextualized representations changes across contexts and words. That is, the division of labor between expectations and lexical meaning appears to be dynamic, shifting as a function of the context and word in question. These trends hold for the two architectures and modes of combination we studied. However, differences among them indicate that a language model's quality influences the degree to which one source is preferred over the other, with the higher quality language model relying more on expectations.

We see two major venues for future research. The first concerns extensions of our framework, in particular aimed at understanding which properties of expectations and words can be used to predict the weight of the contribution of each information source (α). The second concerns further evaluations of our approach. On the one hand, we plan to explore its explanatory breadth as a processing model, using unidirectional language models and data related to online processing (such as reading times and ERP amplitude effects). On the other hand, while our model relies on a single mechanism for interpretation (*avg* or *delta* operations), we are interested in the extent that this can account for different types of ambiguity resolution. In particular, it could be interesting to establish the explanatory capacity of this framework by breaking down its performances into different interpretation tasks, from inferring the correct part of speech of a word to understanding figurative uses of language.

[5]We release the code to use our framework with the LSTM and BERT language models:
https://github.com/amore-upf/exp_lex_interpretation

Acknowledgements

We thank Kristina Gulordava for her precious contributions to the early stages of this research line. This project has received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme (grant agreement No 715154), and from the Catalan government (SGR 2017 1575). We gratefully acknowledge the support of NVIDIA Corporation with the donation of GPUs used for this research, and the computer resources at CTE-POWER and the technical support provided by Barcelona Supercomputing Center (RES-IM2019-3-0006). This paper reflects the authors view only, and the EU is not responsible for any use that may be made of the information it contains.

References

Aina, L., Gulordava, K., & Boleda, G. (2019). Putting words in context: Lstm language models and lexical ambiguity. In *Proc. of ACL*.

Armeni, K., Willems, R. M., & Frank, S. L. (2017). Probabilistic language models in cognitive neuroscience: Promises and pitfalls. *Neuroscience & Biobehavioral Reviews*.

Boleda, G. (2020). Distributional semantics and linguistic theory. *Annual Review of Linguistics*.

Camacho-Collados, J., & Pilehvar, M. T. (2018). From word to sense embeddings: A survey on vector representations of meaning. *Journal of Artificial Intelligence Research*.

Casasanto, D., & Lupyan, G. (2015). All concepts are ad hoc concepts. In *The conceptual mind: New directions in the study of concepts.*

Cosentino, E., Baggio, G., Kontinen, J., & Werning, M. (2017). The time-course of sentence meaning composition. N400 effects of the interaction between context-induced and lexically stored affordances. *Frontiers in Psychology*.

Cruse, D. A. (1986). *Lexical semantics*.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL.*

Erk, K., & Padó, S. (2008). A structured vector space model for word meaning in context. In *Proc. of EMNLP.*

Falkum, I. L., & Vicente, A. (2015). Polysemy: Current perspectives and approaches. *Lingua.*

Filik, R., & Leuthold, H. (2008). Processing local pragmatic anomalies in fictional contexts: Evidence from the N400. *Psychophysiology*.

Fine, A. B., Jaeger, T. F., Farmer, T. A., & Qian, T. (2013). Rapid expectation adaptation during syntactic comprehension. *PloS one*.

Foraker, S., & Murphy, G. L. (2012). Polysemy in sentence comprehension: Effects of meaning dominance. *Journal of Memory and Language*.

Frank, S. L., Otten, L. J., Galli, G., & Vigliocco, G. (2013). Word surprisal predicts N400 amplitude during reading. In *Proc. of ACL.*

Frisson, S. (2009). Semantic underspecification in language processing. *Language and Linguistics Compass*.

Goodman, N. D., & Frank, M. C. (2016). Pragmatic language interpretation as probabilistic inference. *Trends in cognitive sciences*.

Grice, P. (1975). Logic and conversation. *Syntax and Semantics*.

Gulordava, K., Aina, L., & Boleda, G. (2018). How to represent a word and predict it, too: Improving tied architectures for language modelling. In *Proc. of EMNLP.*

Günther, F., Rinaldi, L., & Marelli, M. (2019). Vector-space models of semantic representation from a cognitive perspective: A discussion of common misconceptions. *Perspectives on Psychological Science*.

Huang, Y., & Rao, R. P. (2011). Predictive coding. *Wiley Interdisciplinary Reviews: Cognitive Science*.

Inan, H., Khosravi, K., & Socher, R. (2017). Tying word vectors and word classifiers: A loss framework for language modeling. In *Proc. of ICLR.*

Kao, J., Bergen, L., & Goodman, N. (2014). Formalizing the pragmatics of metaphor understanding. In *Proc. of CogSci.*

Kremer, G., Erk, K., Padó, S., & Thater, S. (2014). What substitutes tell us-analysis of an "all-words" lexical substitution corpus. In *Proc. of EACL.*

Kutas, M., DeLong, K. A., & Smith, N. J. (2011). A look around at what lies ahead: Prediction and predictability in language processing. In *Predictions in the brain.*

Landauer, T. K., & Dumais, S. T. (1997). A Solution to Plato ' s Problem : The Latent Semantic Analysis Theory of Acquisition , Induction , and Representation of Knowledge. *Psychological Review*.

Levy, R. (2008). Expectation-based syntactic comprehension. *Cognition*.

Linzen, T., Dupoux, E., & Goldberg, Y. (2016). Assessing the ability of LSTMs to learn syntax-sensitive dependencies. *Transactions of the ACL.*

McCarthy, D., & Navigli, R. (2007). Semeval-2007 task 10: English lexical substitution task. In *Proc. of workshop on semantic evaluations.*

Mikolov, T., Yih, W.-t., & Zweig, G. (2013). Linguistic Regularities in Continuous Space Word Representations. In *Proc. of NAACL.*

Nieuwland, M. S., & Berkum, J. J. A. V. (2006). When peanuts fall in love: N400 evidence for the power of discourse. *Journal of Cognitive Neuroscience*.

Partee, B. (1995). Lexical semantics and compositionality. *Language*.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proc. of NAACL.*

Pickering, M. J., & Garrod, S. (2007). Do people use language production to make predictions during comprehension? *Trends in Cognitive Sciences*.

Press, O., & Wolf, L. (2017). Using the output embedding to improve language models. In *Proc. of EACL.*

Pustejovsky, J. (1995). *The generative lexicon: A theory of computational lexical semantics.*

Recanati, F. (2004). *Literal meaning*.

Rodd, J. M. (2020). Settling into semantic space: An ambiguity-focused account of word-meaning access. *Perspectives on Psychological Science*.

Smith, N. J., & Levy, R. (2013). The effect of word predictability on reading time is logarithmic. *Cognition*.

Thater, S., Dinu, G., & Pinkal, M. (2009). Ranking paraphrases in context. In *Proc. of TextInfer.*

Wilson, D., & Carston, R. (2007). A unitary approach to lexical pragmatics: Relevance, inference and ad hoc concepts. In *Pragmatics.*

Wilson, D., & Sperber, D. (2006). Relevance theory. *The Handbook of Pragmatics*.

Zarcone, A., Van Schijndel, M., Vogels, J., & Demberg, V. (2016). Salience and attention in surprisal-based accounts of language processing. *Frontiers in Psychology*.