NONLITERAL SEMANTIC EDGE PROBING: STRUCTURE IN CONTEXTUAL WORD EMBEDDINGS

Alex Liebscher
University of California, San Diego
La Jolla, CA 92021
aliebsch@ucsd.edu

December 12, 2019

ABSTRACT

The introduction of contextual word embedding (CWE) models has led to improvements on a wide variety of tasks. Yet, the black-box nature of deep learning language models may be inhibiting further progress. Tenney et al [1] introduced a novel edge probing framework to explore the syntactic and semantic information encoded within contextual embeddings. They assessed the degree to which these types of information are encoded in the embeddings through a series of traditional linguistic tasks. Here, we expand this framework and study how nonliteral meaning may be also encoded within these embeddings. Nonliteral meaning is often highly abstract, conceptual, and cultural. We find that contextual embeddings do encode some level of nonliteral meaning, as distinguished by our probing of metaphor and metonymy detection tasks.

1 Introduction

Recent advances in machine translation and language modeling have produced state-of-the-art contextual word embedding (CWE) models, such as ELMo [2] and BERT [3]. Explaining these models and their embeddings has been an outstanding issue in the field. Attempts have been made to illuminate which, if any, linguistic information is encoded in individual embeddings [1, 4]. They find that embeddings do contain some syntactic information, and to a lesser degree, semantic information. For example, they find CWEs informatively encode constituent roles (i.e. noun phrase, verb phrase, etc.), and somewhat encode semantic relations (ENTITY-ORIGIN, CONTENT-CONTAINER, etc.). In particular, they developed a novel edge probing framework to systematically study types of encoded information within token spans and pairs of token spans. In general, the framework involves using a pretrained language model to encode sentences, then extract token spans and use only these token spans to classify the label or relationship within the sentence. The particularly insight methodology here comes from task classification using *only* the extracted spans. This limits the model to using only information encoded within the token span embeddings, leaving aside the rest of the sentence. Should the embeddings contain sufficient information, they will be enough to perform successfully at a given linguistic task.

However, people are capable of extracting meaning from a nearly boundless number of linguistic phenomenon. In particular, our ability to abstract and create nonliteral language represents a profound mechanism for conceptualization, categorization, and information processing. Knowing how language models encode abstract information may shed light on the mechanics of previous applications of language models on tasks like metaphor detection [5].

2 Edge Probing

The edge probing framework focuses on labeling tasks, not identification tasks [1]. Sentences were encoded with one of the major CWE models: CoVe, ELMo, BERT, and GPT2. The edge probing model uses only the span representations of interest to complete the tasks. This implies that the only information from the sentence that the model may access is that provided by the embeddings of the span. These extracted span representations are input into a small multilayer perceptron (MLP) trained to predict a linguistic task. High performance on these linguistic tasks suggests that contextual embeddings encode information about the task in the span(s) of interest, as opposed to the static noncontextual embeddings previous used.

2.1 Tasks

[1] performed a wide variety of edge probing tasks. They examine part-of-speech tagging, constituent labeling, dependency labeling, named entity recognition, semantic role labeling, coreference, semantic proto-role labeling, and relation classification. Some of these (e.g. part-of-speech tagging) are primarily syntactic tasks, whereas others (e.g. named entity recognition) are tasks primary focused on the underlying semantic information of the target.

To align our results with the original authors', we examine coreference resolution and relation classification. Additionally, we study two new tasks:

Metaphor detection is the semantic task of determining whether a token span is metaphorical or not. We let $s_1 = [i, j)$ be a potentially metaphorical token span within a sentence, and seek to predict Metaphorical or Non-metaphorical.

Metonymy detection is the semantic task of determining whether a token span is metonymic or not. Metonymy is formally "a figure of speech consisting of the use of the name of one thing for that of another of which it is an attribute or with which it is associated" [6]. We let $s_1 = [i, j)$ be a potentially metonymic token span, and seek to predict Metonymic or Non-metonymic.

2.2 Data

Our original intention was to replicate the tasks performed by [1], however the datasets used were largely inaccessible. Because of the difficulty of constructing datasets of these types, there are very few publicly and freely available. Section 4 reveals the implications of this.

Here, we have resorted to using the Definite Pronoun Resolution (DPR) dataset [7]. This dataset follows the Winograd schema and poses a challenge due to its examples' inherent ambiguity.

We also use the relation classification dataset from SemEval 2010 Task 8 [8]. This set consists of English sentences labeled as one of nine directional relationships, or an *other* relation.

The metaphor dataset comes from the TroFi automatic metaphor detection experiment [9]. The data are setup to consist of sentences featuring one of 50 known metaphorical keywords. We include all keywords but only sentences annotated with a known metaphoricity rating.

For metonymy, we use the large metonymy dataset complied by [10]. This set consists of a collection of several metonymy datasets, and an original Wikipedia-based dataset. These data are tailored toward geographic and geopolitical metonymy resolution.

3 Experiments

Our experiments are concerned with contextual word embedding's ability to encode nonliteral meaning. Knowing how embeddings encode nonliteral information may help us interpret the claims that [1] made, that language models encode less semantic information than syntactic information. Some forms of nonliteral language that the edge probing framework may test include metaphor, metonymy, irony, hyperbole, and idiom. Due to time restrictions and lack of accessibility, we were unable to replicate the other tasks the authors probe.

From [1], we assume that CWEs encode more informative representations than baseline, noncontextual embeddings. How well, then, do these embeddings capture even more abstract and conceptual features of language? FOr each example, we ablate a majority of the sentence, leaving all but the tokens of interest for the tasks, and attempt to recover the tasks' targets. Given more time and resources, we would have wished to also compare baseline models for each.

3.1 Representation Models

We explore three successful language models in this edge probing framework: BERT, BERT Large, and OpenAI's GPT2. Each model first tokenizes a string sentence, s, as a list of tokens, $[t_1, t_2, ..., t_k]$. These tokens are input into each model and a list of contextual embeddings, $[e_1, e_2, ..., e_k]$, is produced.

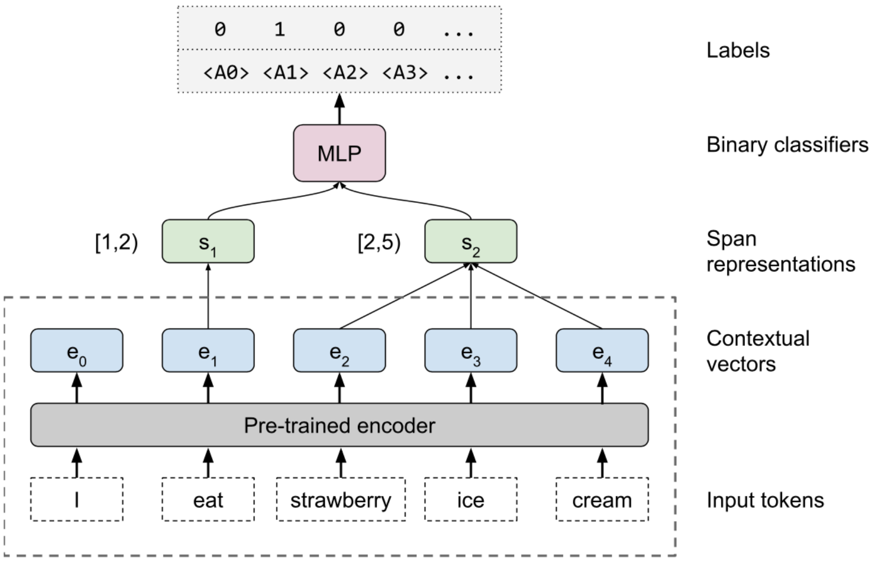

Figure 1: A conceptual visualization of the edge probing model framework. Beginning with an input sentence, a tokenizer creates input tokens. Using a pretrained encoder (e.g. BERT), the tokens from the entire input are embedded into a latent embedding space. Each task specifies spans of importance, for example the object and the pronoun in a coreference resolution task. These embedded token spans $e_i, ..., e_{i+m}$ and $e_j, ..., e_{j+k}$ (if a second span exists) are transformed and used as input into a small multilayer perceptron with the object of predicting the task label or relationship.

BERT/BERT Large [3] are 12 and 24-layer Transformer [11] encoders, pre-trained on unlabeled data over two different pre-training tasks and fine-tuned over other labeled tasks. Importantly, these two pre-training tasks include a masked language model approach and a next sentence prediction task.

OpenAI GPT2 [12] is a 12-layer Transformer [11] encoder trained left-to-right on input text to faciliate text generation. The model is trained over 8 million documents of diverse text.

3.2 Edge Probing Model

For each task, we begin by embedding the entire sentence using a language model. Each task uses either one target span or two. The extracted span embeddings are then linearly projected into a lower dimension and are followed by a pooling operator. Pooling is only done within the spans themselves and the remaining sentence context is not incorporated in the model after this point.

The span representations are concatenated if more than one exist. These are then input into a two-layer MLP followed by a sigmoid output layer. A simple architectural depiction may be found in Fig 1. Binary cross-entropy was used during training for the target label set $|\mathcal{L}| \in \{0, 1\}$.

40 epochs of training were performed for each task and each model with a batch size of 32. It is unclear how many epochs the original authors train over. A validation set was tested every five training epochs. Like the original work, a learning rate scheduler reduced the learning rate to one-tenth if the validation set loss failed to improve for 2 validation steps. The original authors halved the learning rate, however due to differences in data, reducing to one-tenth provided better results for us.

4 Results

After training, we take the best fit from each task and model, and evaluate a held-out test set. We report F1 scores for these evaluations in Table 1. Immediately, we observe large within-task variation, which differs from [1], who found within-task metrics didn't differ by more than about 0.1 F1 points. Nonetheless, it is apparent that the contextual embeddings encoded some information within spans for completing nonliteral tasks. GPT2 appeared to encode the least information, which may be explained by the model architecture focusing only on leftward context. Nonliteral language may make heavier use of the surrounding context and hence, we see GPT2 perform an average of 0.1 and 0.08 F1 points lower than BERT and BERT Large, respectively. Compare to what [1] found, where GPT2 performed 0.02 and 0.01 F1 points lower than BERT and BERT Large, respectively. The training and validation loss for the BERT encoder over each task may be found in Fig 2.

	BERT	**BERT Large**	**GPT2**
Relation	0.674	0.606	0.529
Coref.	0.388	0.484	0.369
Metaphor	0.751	0.675	0.619
Metonymy	0.885	0.855	0.801
Macro Average	0.675	0.655	0.580

Table 1: F1 scores across tasks and models, including macro average scores per model.

Like [1], we see poor performance on the Winograd coreference task. Again, this is a semantically complex task as noted by the original authors. The Winograd schema focuses on ambiguous pronoun resolution and therefore would require more detailed inferential capabilities to do well.

Overall, the results for relation classification and coreference tasks were lower than the original study. This could be the case for several reasons. First, some aspects of the original edge probing model were left unimplemented to save time and resources. For example, [1] use a self-attentional pooling operator over token spans, whereas we use a max pooling operator. A slightly simpler token alignment algorithm was used here, although qualitative examination of the tokenization process suggests there was no significant difference. We also use a simple concatenation of subword embeddings, which does not include top layer activations in any account.

Lastly, it is likely that there was insufficient data to fully understand the mechanisms underlying the probing here. Our largest dataset was for the metonymy task, which makes it difficult to fully compare results between studies. Additionally, the literal data we procured compared to the nonliteral datasets is different both in terms of number of target labels and number of spans used. Despite complications arising from this difference, it's not completely unreasonable to extrapolate the results and make broader comparisons.

We do notice that contextual word embeddings appear to contain enough information to achieve respectable metrics on both the metonymy and metaphor tasks. This provides hopeful evidence that a more comprehensive study would make a better performance comparison between the highly conceptual tasks and the syntactic tasks.

5 Conclusion

Overall, despite discrepancies in the current study and [1], we find that contextual word embeddings models encode, to a limited degree, nonliteral semantic information. An important step was taken here to address the black-box nature of these complex Transformer-based language models. Mainly, we ask and suggest a solution for the extent to which contextual word embeddings encode highly conceptual and abstract linguistic information. These models have performed well on tasks such as metaphor detection [5], and probing the structure of the embeddings used may help explain these results. We see here that success on tasks such as metaphor detection may certainly use information encoded in the keywords of interest, as opposed to other information encoded in surrounding contexts.

A Appendix

Table 2: Task dataset statistics are described here, including the number of target labels, the number of spans extracted per sentence, total examples (train/dev/test), and total tokens.

Task	$|\mathcal{L}|$	Spans	Examples	Tokens
Relation	19	2	6.8k / 1.2k / 2.7k	128k / 23k / 51k
Coref.	1	2	1.4k / 170 / 700	20k / 10k / 3.5k
Metaphor	1	1	2.1k / 700 / 900	60k / 20k / 27k
Metonymy	1	1	9k / 2.3k / 3.8k	246k / 61k / 104k

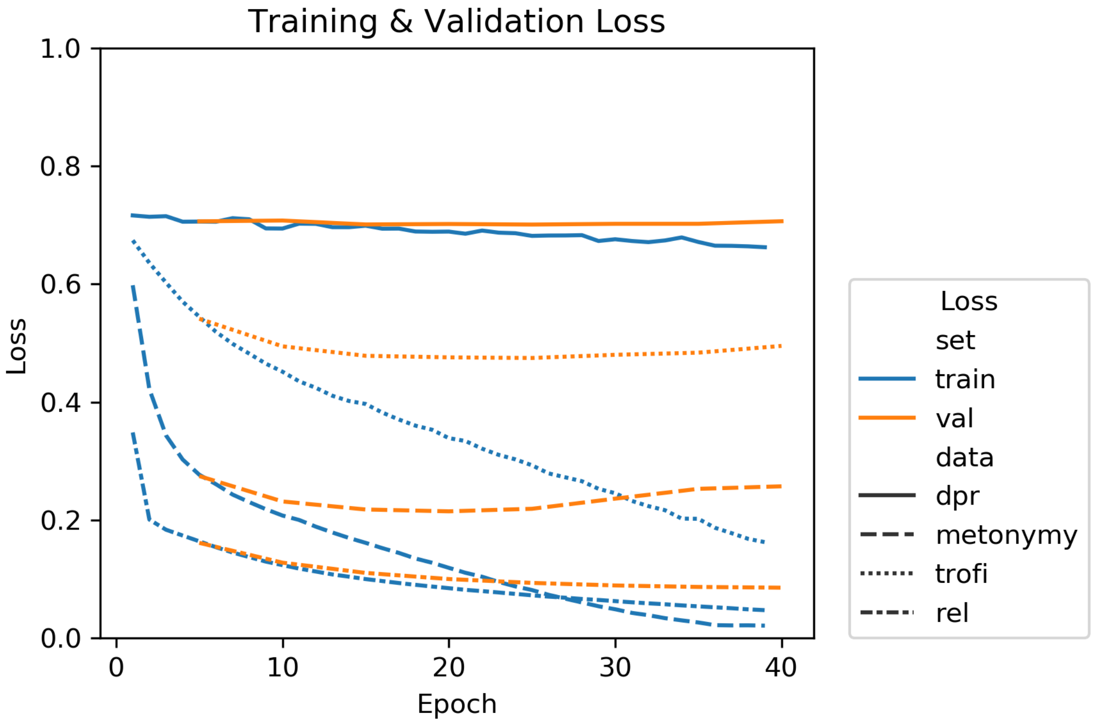

Figure 2: Training and validation loss for BERT base across four tasks. Notice the poor performance on the coreference task (see Sec 4)

References

[1] 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. What do you learn from context? probing for sentence structure in contextualized word representations. *arXiv preprint arXiv:1905.06316*, 2019.

[2] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[4] Ian Tenney, Dipanjan Das, and Ellie Pavlick. Bert rediscovers the classical nlp pipeline. *arXiv preprint arXiv:1905.05950*, 2019.

[5] Rui Mao, Chenghua Lin, and Frank Guerin. End-to-end sequential metaphor identification inspired by linguistic theories. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3888–3898, 2019.

[6] The Merriam-Webster.com Dictionary. Metonymy. Retrieved December 10, 2019.

[7] Aaron Steven White, Pushpendre Rastogi, Kevin Duh, and Benjamin Van Durme. Inference is everything: Recasting semantic resources into a unified evaluation framework. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 996–1005, Taipei, Taiwan, November 2017. Asian Federation of Natural Language Processing.

[8] Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden, 2010. Association for Computational Linguistics.

[9] Julia Birke and Anoop Sarkar. A clustering approach for nearly unsupervised recognition of nonliteral language. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, 2006.

[10] Milan Gritta, Mohammad Taher Pilehvar, Nut Limsopatham, and Nigel Collier. Vancouver welcomes you! minimalist location metonymy resolution. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1248–1259, 2017.

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[12] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8), 2019.