Sentence-Level Semantic Probing: A Literature Review

Zhejian Peng
zpeng94@stanford.edu
Riccardo Melucci
rmelucci@stanford.edu
Timo Wang
ningwang2023@u.northwestern.edu
Cameron Thouati de Tazoult
cameron8@stanford.edu

1 Introduction

For the final project, we propose to analyze vectorized sentence representations produced by various language models. We hypothesize that there exists a vector subspace in the union of each language model that captures more semantic information than each original model's representations do on their own. In this preliminary paper, we review nine related papers that study the robustness and effectiveness of language models from different perspectives. We first discuss how well language models are capable of encoding syntactic and semantic information, where some use the probing technique. Then we briefly discuss one task that cannot be easily solved by language models and argue why failure to solve such a task does not weaken our motivation on studying the semantic expressiveness of language models. After that we proceed to review papers on the designing and interpretation of probes. Lastly we discuss two existing benchmarks for NLU models that are potentially useful for evaluating and building our probes.

2 Representation Capabilities of Language Models

Hewitt and Manning (2019) study whether representations produced by deep models encode syntactic information. Specifically, they explore the potential of deep contextual models encoding entire parse trees in their word representations by proposing a *structural probe*, a simple model testing whether such word representations have certain linear transformations that consistently correspond to the structure of syntax trees. This correspondancy is determined through a proportional relationship between the squared L2 distance between the transformed embedding vectors of two words and the number of edges between these two words in the parse tree. In their experiments, they compare how well different models are capable of encoding syntactic information by performing an ablation study involving variations of ELMo and BERT as well as several baseline models. They find out that such deep language models as ELMo and BERT are robust at embedding syntax while the majority of baseline models fail to encode meaningful syntactic information. This is expectable because whereas ELMo and BERT can capture and utilize deep correlation between words spanning across potentially long texts, most baseline models do not utilize any contextual information or make strong assumptions about the structure of English parse trees. They also find out that the rank of the linear transformation used to obtain vector subspace containing syntactic information is surprisingly low: increasing the rank beyond 64 does not improve the parsing accuracy. Furthermore, they note that both ELMo and BERT require transformations of roughly the same rank. This observation is interesting, because it indicates the potential existence of a similar transformation with which we can obtain a vector subspace encoding rich semantic information. It is also of interest to study whether the semantic information can be encoded in a relatively smaller vector space and, if yes, explores whether the dimension size of the original representations can be significantly reduced to make the model more portable and easily trained.

Coenen, Reif, Yuan, Kim, Pearce, Viégas, and Wattenberg (2019), take a similar direction to the

one of structural probing (Hewitt and Manning, 2019). However, instead of further studying a syntactic subspace of the word representations generated by BERT, they study the effectiveness of the attention matrices representing dependency information. Specifically, they train a linear model to classify a given relation between two input tokens. If the linear model is able to achieve reliable accuracy, and it does, it is safe to conclude that the attention matrices are able to encode enough dependency information. Furthermore, they also provide a mathematical explanation for why squared Euclidean distance for parse tree distance, proposed by Hewitt and Manning (2019), is reasonable and build a tool to visualize the tree structure embedded in the linearly transformed word embedding subspace. In addition to studying the BERT's capability of encoding syntactic information, they explore semantic information stored within word-level embeddings. Each word has a contextualized embedding produced by BERT, and the information may vary from sentence to sentence. Therefore, the embedding vectors for a word can be from different clusters in the vector space. Combined with a nearest-neighbor classifier, such information can be used for word sense disambiguation. In their experiments, Coenen et al. (2019) discover that BERT is capable of capturing sufficient semantic information that the test of word disambiguation results in a F1 score over 70. Moreover, they perform a word-level semantic probing and, with the trained linear model, are able to increase the F1 score from 71.1 to 71.5. The work done by Hewitt and Manning (2019) and Coenen et al. (2019) shows promising signs that word-level information encoded by such language models as ELMo and BERT can be further purified for particular tasks. We wonder if the sentence-level information, such as sentence embeddings produced by BERT, can also be further processed into separate syntactic and semantic subspaces.

Tenney, Das, and Pavlick (2019) study if specific types of linguistic information are encoded through the sentence-level encoders, such as ELMo and BERT. This helps understand whether such models learn meaningful abstractions for representing natural language or just statistical metrics. Observations shows syntactic and semantic information are encoded at different layers of language model. The further this idea and show how individual sentences are encoded by the BERT model, and show different layers contain different sentence information through analyzing the results of eight classification tasks. The experiments use *edge probing* to measure how well linguistic information is represented through BERT's pre-trained encoder. A sentence is presented as a list of tokens and part of the sentence forms a training sample $\{s_1, s_2, L\}$, where L can be multiple labels. A probing classifier model is trained with only input of per-token contextual vectors, so the model must rely on the encoder to provide syntactic and semantic information. Eight labels include, part-of-speech (POS), constituents (Consts.), dependencies (Deps.), entities, semantic role labeling (SRL), coreference (Coref.), semantic proto-roles (SPR), and relation classification (SemEval). Classification results are evaluated with micro-averaged F1. Two metrics are defined to analyze different linguistic information in each layer. Scalar mixing weights are used to assess the importance of each layer of the BERT model. Cumulative scoring shows incremental score change introduced by each layer. Scalar mixing weights are learned through training data, and cumulative scoring is computed from the evaluation set. Through analysis, they discover the lower layers contain more syntactic information while the higher layers contain more high-level semantic information. Additionally, syntactic information is more localized while semantic information exists across the network. They also show that the probes can revise their classification decisions after more layers of information been added. Overall, Tenney et al. (2019) shows evidence that deep language models can abstract syntactic and semantic information from sentences.

3 Downstreaming Task Solving based on Language Models

While many tasks such as Argument Reasoning Comprehension appear to be solved by BERT with near-human accuracies, even without the required world knowledge for this task, Niven and Kao (2019) show that BERT is capable of achieving such tasks not by truly understanding the text content and then making correct inferences but rather by exploiting spurious statistical cues in the dataset. They demonstrate this by constructing an adversarial dataset, on which all tested models achieved random accuracy. However, this conclusion is expected, because the nature of deep learning models

is to capture useful correlations lying within the dataset and encode them based on different tasks. Such exploitation of correlation accounts for their capacity to classify text or uncover underlying patterns within data. Therefore, performing poorly on tasks that involve logical reasoning is thus not a necessary proof that language models like BERT cannot encode key semantic information. However, it would be interesting to explore the possibility of combining deep models' capability of recognizing patterns and ontology-based inference engines' capability of inferring possible outcomes in order to attack tasks that require logical reasoning skill and language understanding ability.

4 Designing and Interpreting Probes

Probes are supervised models trained to learn various linguistic tasks and achieve high accuracy. Hewitt and Manning (2019) use *structural probe* to study linear transformations between contextual word representations and the structure of syntax trees. Tenney et al. (2019) also use *edge probing* to study what linguistic information is encoded through sentence-level encoders. Well designed probes can provide better interpretability of deep language models and achieve higher accuracy on downstream linguistic tasks. Hewitt and Liang (2019) present *control tasks* against other linguistic tasks to select good probes and claim good probes should achieve high linguistic tasks accuracy but low control tasks accuracy. Control tasks associate word types with random output, so they can only be learned if the probe memorizes it. They define selectivity as the accuracy difference between linguistic tasks and control tasks. Good probes should achieve high accuracy on linguistic tasks but low accuracy on control tasks, thus result in high selectivity.

They train probes on two classification tasks: part-of-speech and tagging, and dependency edge prediction. For the part-of-speech tagging task, they experiment with linear, MLP-1, and MLP-2 probes with softmax as the output layer and ReLU as the activation function. For dependency edge prediction, they experiment with bilinear, MLP-1, and MLP-2 probes. They also experiment with 5 complexity control methods: rank/hidden dimensionality constraint, dropout, training data size constraint, weight decay, and early stopping. Dropout and early stopping don't improve selectivity, while the other three improve selectivity without largely affecting linguistic task accuracy. For MLP probes, experiments show constraining the hidden state dimension improves selectivity at little cost to linguistic task accuracy. MLP Hidden sizes of 10 (part-of-speech tagging) and 50 (dependency edge prediction) result in high selectivity while maintaining linguistic task accuracy. Thus, MLP with higher hidden size is generally overparameterized. Constraining training data size is effective for part-of-speech tagging task, indicating that learning linguistic tasks need fewer data than control tasks. Out of all the probes, they find linear and bilinear probes show the best selectivity without the complexity control method, but the most accurate probes are MLPs, especially for dependency edge prediction. This suggests that linear models are not the best at learning syntactic tree information. Thus for some linguistic tasks, MLP probes with hyper-parameters tuned for selectivity are better at extracting non-linear features.

Yaghoobzadeh, Kann, Hazen, Agirre, and Schutze (2019) had similar findings regarding non-linear probes when exploring if multiple senses of a word are represented in single vector word embeddings. The team built two probing tasks, an S-class prediction task and an ambiguity prediction task. In the former, they trained binary classifiers that take an embedding as input and predict what s-class it belongs to. In the latter, they trained the classifiers to predict whether an embedding is ambiguous (belongs to more than one S-class) or not (only belongs to one). These probes were used on WIKI-PSE: a large S-class/word pair dataset that they created from manual Wikipedia annotations and word senses, as the other existing sense corpora such as WordNet and SemCor were too small for their intended use. This database will be very valuable when we making decisions for modeling tradeoffs regarding words with ambiguous senses.

For the s-class prediction task, they experimented with three classifiers: (i) logistic regression (LR), (ii) multi-layer perceptron with one layer hidden and a final ReLU layer, and (iii) KNN: K-nearest neighbors. They found that MLP consistently outperformed the other classifiers; this suggests that the embedding space is not linearly separable with respect to the S-classes, confirming that linear classifiers are also insufficient for semantic probes. This helps us target our efforts towards

more powerful non-linear classifiers when thinking about semantic encodings. They also found that as long as a sense was frequent enough in the training dataset, the standard word embeddings are able to capture multiple senses in their single vector format well. This is helpful towards our decisions for what embedding models to consider, as we see that including word order in a model is important for a robust representation. Lastly, Yaghoobzadeh et al. (2019) also found that higher dimensional embedding spaces allowed for better capturing difficult cases of ambiguity, but made cosine similarity less predictive of S-class. This will likely be a trade-off we will encounter when deciding how we will encode sense information based on context for our project, and the WIKI-PSE database will be very valuable in making such decisions.

5 Existing Benchmarks for NLU Models

As noted in the introduction by Dasgupta, Guo, Stuhlmuller, Gershman, and Goodman (2018), the choice of benchmark datasets is crucial to understanding whether a model's performance is thanks to complex semantic information being encoded into the model or simply due to the structure/bias of the dataset. Most current models are only able to keep word-level representations: they cannot account very well for compositionality which is one of the keys to understanding human language. The compositional comparisons dataset would be useful for ensuring that the models we explore for sentence representations are actually encoding compositional information, and thus building a transferable model that can be generalized to several task contexts.

In their aforementioned study, Niven and Kao (2019) were able to produce an adversarial dataset that fooled models into achieving random accuracy. This is not an all-encompassing metric to determine the extent to which semantic information is being captured by a model. However, some publicly available benchmarks exist that measure model performance on various tasks. Conneau and Kiela (2018) present one such benchmark – SentEval. SentEval evaluates models based on a set of predefined classification tasks, providing a more detailed analysis than Niven and Kao (2019); determining which kinds of specific tasks a model performs well or poorly on can provide a better picture of what kind of information the model captures. SentEval evaluates models on 17 downstream tasks and 10 probing tasks involving binary and multi-class classification, entailment and semantic relatedness, semantic textual similarity, paraphrase detection, and caption-image retrieval. This open-source evaluation tool could serve us for our project as it is intentionally very easy to use and could provide a good overview analysis of a model to better inform any semantic probing we may want to perform. Additionally, it serves as a good reference for community-accepted probing tasks that we may want to modify for our probing. Several of the downstream tasks involving semantic similarity could also be useful.

SentEval provides several useful benchmarks for evaluating sentence representations from NLU models, but Wang, Singh, Michael, Hill, Levy, and Bowman (2019) point out multiple ways in which SentEval falls short. First, several of the classification tasks used by SentEval are either solved or very close to being solved, rendering them fairly uninformative for model benchmarking. Moreover, SentEval only evaluates sentence-to-vector encoders and is better for evaluating sentences by themselves without additional context. Wang et al. (2019) propose a new benchmarking platform called GLUE that fixes these flaws. GLUE incorporates more difficult classification tasks than SentEval and is model-agnostic, meaning it can evaluate any model that can make predictions based on a sentence or sentence-pair. This allows for any kind of sentence representation or contextualization whether or not the embedding comes in the form of a vector. GLUE's main benchmark evaluates on 9 separate NLU tasks that focus on practical model applications, but GLUE also includes a diagnostic dataset that annotates many types of specific linguistic phenomena and could be used for semantic probing as well as development of adversarial examples.

While the GLUE benchmark is both more difficult and makes use of more complex evaluations than SentEval, both tools could prove useful to our project. Analyzing the results of these tool can help inform us about what specific tasks models fail at, in turn informing the designs of our semantic probes. Additionally, the inner workings of the tools themselves could be analyzed as they both use probing techniques that may be incorporated into our designs.

6 Conclusions and Future Work

This literature review serves as an overview of current work in the field of NLU model semantic probing. While we have evidence that structural probing techniques can be very effective at determining how models capture syntactic information, work on semantic probing is sparser and less conclusive. However, both NLU model benchmarking tools and individual studies of semantic probing have some overlap in their basic methodology; the differences between their specific classification tasks allow room for exploration in promising directions for our project. We aim to find an effective transformation from contextual word representations (such as those provided by BERT) to sentence-level semantic representations. In addition, we plan to use benchmark databases such as the compositional comparisons dataset by Dasgupta et al. (2018), the WIKI-PSE dataset by Yaghoobzadeh et al. (2019), and the dataset used in Niven and Kao (2019) to probe different aspects of our model's information encoding.

References

Andy Coenen, Emily Reif, Ann Yuan, Been Kim, Adam Pearce, Fernanda Viégas, and Martin Wattenberg. 2019. Visualizing and measuring the geometry of bert.

Alexis Conneau and Douwe Kiela. 2018. Senteval: An evaluation toolkit for universal sentence representations.

Ishita Dasgupta, Demi Guo, Andreas Stuhlmuller, Samuel J. Gershman, and Noah D. Goodman. 2018. Evaluating compositionality in sentence embeddings.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks.

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

Timothy Niven and Hung-Yu Kao. 2019. Probing neural network comprehension of natural language arguments.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. Bert rediscovers the classical nlp pipeline.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. Glue: A multi-task benchmark and analysis platform for natural language understanding.

Yadollah Yaghoobzadeh, Katharina Kann, Timothy J. Hazen, Eneko Agirre, and Hinrich Schutze. 2019. Probing for semantic classes: Diagnosing the meaning content of word embeddings.