Analyzing BERT's Knowledge of Hypernymy via Prompting

Michael Hanna and **David Mareček**
Institute of Formal and Applied Linguistics, Faculty of Mathematics and Physics
Charles University, Prague, Czech Republic
hannami@o365.cuni.cz, marecek@ufal.mff.cuni.cz

Abstract

The high performance of large pretrained language models (LLMs) such as BERT (Devlin et al., 2019) on NLP tasks has prompted questions about BERT's linguistic capabilities, and how they differ from humans'. In this paper, we approach this question by examining BERT's knowledge of lexical semantic relations. We focus on hypernymy, the "is-a" relation that relates a word to a superordinate category.

We use a prompting methodology to simply ask BERT what the hypernym of a given word is. We find that, in a setting where all hypernyms are guessable via prompting, BERT knows hypernyms with up to 57% accuracy. Moreover, BERT with prompting outperforms other unsupervised models for hypernym discovery even in an unconstrained scenario. However, BERT's predictions and performance on a dataset containing uncommon hyponyms and hypernyms indicate that its knowledge of hypernymy is still limited.

1 Introduction

Large pretrained language models (Devlin et al., 2019; Radford et al., 2019) have set new standards for performance on NLP tasks. Many of these tasks, such as question answering and natural language inference, might seem to require human-like syntactic and semantic capabilities to perform well, leading to many studies on this topic (Rogers et al., 2020).

However, evidence that LLMs have human-like semantic capabilities is mixed. With respect to polysemy, Wiedemann et al. (2019) find that the sense of a polysemous word can be disambiguated using a k-nearest-neighbor approach on BERT's representation of the word. In contrast, Yenicelik et al. (2020) note that while polysemous words' senses are linearly separable based on their BERT embeddings, these embeddings do not form clusters based on their senses alone.

In terms of BERT's knowledge of lexical semantics, Vulić et al. (2020) find that BERT's embeddings capture useful lexical-type knowledge, performing better than non-contextual embeddings on a lexical relation prediction task. That said, BERT's knowledge of lexical semantics is not equal for all words: it struggles with rare words (Schick and Schütze, 2020).

In this work, we further examine BERT's knowledge of lexical semantics, more specifically that of hypernymy, using a Cloze / prompting methodology. In the most basic form of this framework, we simply run BERT on an input sentence such as "An apple is a `[MASK]`.", and extract the probabilities assigned to each token that could fill in the blank; here, we would expect an answer such as "fruit".

While Ravichander et al. (2020) also approach BERT's lexical semantics using a prompting task, they focus on simple prompts and examine only if BERT can predict a word's canonical hypernym. In contrast, we explore how more complex prompts affect BERT's ability to predict hypernyms, propose new methods of evaluating hypernym discovery that that take into account the fact that a word may have multiple hypernyms, and find effective prompts for hypernym discovery.

2 Background: Prompting

Although fine-tuning is the methodology with which LLMs shattered existing state-of-the-art on many downstream tasks, the use of prompting has also gained interest in recent years. Unlike fine-tuning, prompting involves no further training to solve downstream tasks; instead, a prompt is fed to the language model, which either predicts the continuation of the prompt (autoregressive language modeling) or the identity of one or more masked words (masked language modeling). The output of the language model is then used as the response to the task at hand.

Prompting has seen use for diverse tasks such as knowledge base completion (Petroni et al., 2019) as well as summarization and translation (Radford et al., 2019). While performance on these tasks is not always up to par with supervised state of the art, prompting has the advantage of being unsupervised. However, various studies have found that task performance is highly dependent on the prompt used (Jiang et al., 2020; Reynolds and McDonell, 2021); these same studies propose automatic or hand-made improvements to prompts to improve them. Supervised techniques have also been devised to automatically develop better prompts (Shin et al., 2020).

Beyond downstream tasks, prompting is also useful for investigating language models as a test subject. While probing (Conneau et al., 2018; Tenney et al., 2019), for example, is useful, it involves training an auxiliary model. This auxiliary model can be troublesome; (Hewitt and Liang, 2019) show that the probes themselves can learn tasks independently of whether the model encodes linguistic structure. Moreover, a model can contain information within its internal representations without relying on that information to make predictions. In contrast, prompting requires no auxiliary models that might complicate matters.

Ettinger (2020) uses prompting to compare BERT's linguistic competencies and tendencies to that of humans; among those investigated is hypernymy. They find that BERT easily identifies noun hypernyms, but fails to adapt to negated sentences: given the prompts "A hammer is a `[MASK]`" and "A hammer is not a `[MASK]`", BERT predicts the words "hammer", "tool", and "weapon" for both.

In a follow-up, Ravichander et al. (2020), investigate whether BERT has a systematic understanding of hypernymy. They find that BERT's understanding of hypernymy is not systematic: BERT fails when tasked with finding the hypernym of a hyponym in the plural; it also fails more often when the hyponym in question is uncommon, or has not been seen with its hypernym in Wikipedia. In contrast to the practically-oriented works, these studies that analyze LLMs use primarily simple prompts to extract BERT's predictions.

These two uses of prompting, for downstream tasks and for analysis of LLMs, are somewhat in tension. On one hand, studies searching for knowledge in BERT treat it as a test subject, and use prompting to search for knowledge of hypernymy using very simple prompts. On the other hand, from practically-oriented studies, we know that using simple prompts may lead to BERT underperforming, if only because it is not responding to the task at hand. The use of simple prompts might thus obscure knowledge of hypernymy in BERT.

In this paper, we attempt to balance these two approaches. We explore more complex prompts, in an attempt to reveal knowledge of hypernymy in BERT that might be hidden by challenges with prompting. However, we limit prompts to those which are human-understandable, and can be created without any supervision (excluding, thus, prompt-tuning approaches). Moreover, we compare BERT's performance on a hypernym discovery task to that of other systems, to give context to its hypernym discovery ability.

3 BERT Diagnostics

In this section, we use prompting to determine how well BERT understands hypernymy. In order to predict the hypernym of a word (hyponym) using BERT and prompting, we run BERT on a prompt including a mask token, using it to predict the identity of the masked token. One such prompt might be "An apple tastes good. An apple is a `[MASK]`.", where the hyponym is "apple", and its canonical hypernym is "fruit."

3.1 Dataset

We perform our experiments on a subset of the hypernyms / hyponyms given by Battig and Montague (1969); we call this the Battig dataset. The dataset consists of pairs (x, y) of hyponym word x and canonical hypernym y; these pairs are organized by y, and many words share the same hypernym. Our subset consists of 863 words that have unique 25 hypernyms in total. We select only words where y would be represented by BERT as one token; doing this allows the hypernyms to be guessed by BERT using one `[MASK]` token.

3.2 Experiments

Using the Battig dataset, we conduct a variety of experiments, to determine to what degree BERT is aware of hypernymy. In each experiment, we fix a given prompt, with a slot for a hyponym, as well as a mask token which will be used to obtain the hypernym. Then, for each hyponym in the Battig datset, we use Wolf et al. (2020)'s `bert-base-uncased` BERT model to obtain the probabilities assigned to each word (potential hypernym). Here, we detail the prompts used; in Section 3.3, we discuss how we evaluate our results.

In our first experiment, the **basic** experiment, we extract hypernyms from BERT using the simple prompt "A(n) x is a `[MASK]`.". We use the appropriate article ("a", "an", or none at all) for x, but we always use the article "a" as the determiner for `[MASK]`. This results in difficulties in guessing hypernyms that start with vowels. Although a prompt such as "xs are `[MASK]`." would avoid this issues, using such a prompt is not possible due to tokenization issues: while all hypernyms in the singular are single tokens, hypernyms in the plural are often split into multiple tokens, and cannot be guessed using one `[MASK]` token.

As a follow up, we run the **type-of** experiment, using the prompt "A(n) x is a type[1] of `[MASK]`.". This is intended to allow for vowel hypernyms and clarify that hypernymy is the relation of interest.

We also test two final simple prompts that reflect common real-world situations in which hypernyms might appear. The first, "a `[MASK]`, such as x", is a prompt first proposed as a template using which hypernym-hyponym pairs could be discovered in unlabeled text (Hearst, 1992). This **such-as** prompt is a fragment, rather than a natural sentence. The second prompt is "My favorite `[MASK]` is x.". This "**favorite**" prompt both provides a more natural prompt, and puts the `[MASK]` token in an intermediate position.

Next, we perform a series of contextualized queries, based on the hypothesis that some hypernyms are difficult for BERT to predict because there is insufficient information in the query, and our short, contextless prompts are too different from the training data. To resolve this we generate contexts in two ways. First, we create a **handwritten context** to be used for each hypernym. For example, if the word x is a flower, its context might be "A(n) x smells nice.", followed by the prompt "A(n) x is a `[MASK]`". Note that since there is no training, BERT cannot simply learn a context-hypernym correspondence. However, although we attempted to write prompts that did not hint very strongly at the identity of the hypernym, it is possible that these contexts might have provided clues.

We also experiment with **automatically gathered contexts**, gathered from SemCor 2.0 (Langone et al., 2004). To do this, we first automatically find the WordNet sense of each word in the dataset. We assign to each word the first WordNet sense (if any) that includes the word's hypernym as an inherited hypernym. Then, we choose as the word's context the first SemCor sentence that includes the word used in the appropriate sense.

Additionally we experiment with prompts that include **multiple hyponyms**. Specifically, we use two prompts: "A(n) x is a `[MASK]`. So is a(n) x'." and "My favorite `[MASK]` is either a(n) x or a(n) x'", where x' is another word, in principle another hyponym. These prompts allow us to query the hypernym of polysemous words more clearly: while the word "orange" admits the hypernyms "color" and "fruit", when "orange" is paired with "purple", only the former is acceptable. In the first experiment, we select x' from the actual pool of words that share x's hypernym.

In the **FastText multiple hyponym** experiment, we automatically find values for x'. We represent each word in the dataset as its FastText (Bojanowski et al., 2016) vector, using the `eng-300` pretrained model. Then, we choose as x' the nearest neighbor (using cosine distance) of the word of interest x within the FastText model's vocabulary. Using this method, we can generate x' while keeping the prompting process entirely unsupervised.

Finally, we test an **ensemble** strategy: we take BERT's predictions based on an unweighted average of three prompts: "x is a `[MASK]`", "x is an `[MASK]`", and "x is a type of `[MASK]`". Since the last of the three prompts gives no hint as to the starting letter of the following word, it acts as a tiebreaker between possible vowel / consonant-initial responses.

3.3 Evaluation

We use a variety of metrics to evaluate the hypernyms predicted by BERT. First, we use **precision at 1** (P@1), that is, the proportion of BERT's most-likely predictions that were correct, relative to the canonical hypernym given by the dataset. We also use **mean reciprocal rank** (MRR), which is the mean (taken over examples) of the reciprocal of the rank of the correct answer in BERT's top predictions. We consider only BERT's top 5 predictions, and assign a reciprocal rank of 0 if the answer is not in the top 5.

However, these metrics are flawed, as many words have multiple hypernyms, possibly corre-

[1] We experiment with other words besides "type" that might indicate hypernymy, such as "kind", "category", and "sort", but find no significant difference in results.

sponding to multiple senses of the word. For example, an oak is a type of tree, but also a type of plant; orange, is both a type of fruit and a color. To resolve this, we also create an automated metric, **WordNet P@k**, to determine if a predicted hypernym is valid. In WordNet (Fellbaum, 1998), most nouns have for each word sense a list of inherited hypernyms, including both its direct hypernyms, as well as the inherited hypernyms of the noun's hypernyms. In our WordNet precision at k (P@k) metric, a word w is considered to be a hypernym of a word x if w is an inherited hypernym of x. WordNet P@k is thus the proportion of the top k predicted hypernyms for x that are one of x's inherited hypernyms.

This metric, too, has flaws; because WordNet's word senses are very fine-grained, WordNet's hypernyms do not correspond directly with intuition, penalizing hypernyms that might be valid. Moreover, in some cases, WordNet's hypernyms for a word do not include the hypernym label from the dataset; e.g. WordNet does not include"boat" as a hypernym for "yacht". To partially resolve this issue, we label a prediction correct under WordNet if the prediction is a hypernym from the dataset or an inherited hypernym according to WordNet.

We also notice that in many cases, BERT's most likely prediction for a given word's hypernym is the word itself. In the case of a prompt such as "x is a [MASK]", the tautology "x is x" is not strictly incorrect. Moreover, it is not uncommon that, when x is predicted as its own hypernym, the second most likely prediction is the canonical hypernym of x. So, we report results both including x as a possible prediction, and excluding x when BERT predicts it as its own hypernym.

3.4 Results

For each of our experiments, we use a different set of prompts to predict the top hypernyms of each hyponym in the dataset, and report four metrics: P@1, MRR, WordNet P@1, and WordNet P@5, as defined above. Table 1 reports these metrics as expected, while Table 2 shows the results predicting the hypernyms for a word x with BERT, and deleting x from the list of hypernyms, as BERT often predicts x to be its own hypernym.

3.5 Discussion

The results indicate that the basic experiments, using only prompts of the form "x is a [MASK]" prompt BERT to give the correct answer only 30% of the time. In contrast, explicitly stating the (type-of) relationship intended results in a precision of only 16.2, likely because these constructions are less common in the training data than those without "type-of". This is despite the fact that this prompt avoids the troubles of using "a/an" in the prompt.

The "such-as" prompt is more effective than the prior two: its precision is 18 points higher than the basic prompt, and its MRR is 6 points higher; moreover, its WordNet P@1 is the highest of all prompts using only one hyponym. The "favorite" prompt also performs favorably compared to the first two, although not as well as the "such-as" prompt.

The addition of handcrafted context to the prompt, intended to better replicate BERT's training conditions and disambiguate the meaning of the hyponym in question, improves BERT's hypernym-guessing ability. However, the automatically-found contexts do not yield the same boost; manual analysis reveals that BERT often guesses words that might reasonably continue the text given the context, but these are not necessarily hypernyms.

Adding another hyponym to the prompt in addition to x is the most effective at improving BERT's accuracy. When both hyponyms share the same hypernym, the WordNet P@1 rises to 60%, which is much higher than the baseline basic prompt, but still demonstrates room to improve. Some of the errors (BERT fails to guess the hypernyms "animal" and "instrument") are due to the hypernym's starting with a vowel; others, like the BERT's inability to guess hypernyms "relative" and "cloth", do not admit such an easy explanation. In the case of "cloth", while in the case of "relative", BERT simply guesses wrong.

Although the multiple hyponym method does not allow BERT to perfectly guess the correct hypernym, its MRR of 66.7 (excluding x) is impressive. Moreover, we can use unsupervised methods to find a second hypernym, with only a moderate loss in accuracy.

Finally, the prompt ensemble technique offers no improvement over the basic prompt in terms of accuracy with respect to the canonical hypernym. However, when excluding x from the predicted hypernyms and measuring using WordNet accuracy, the ensemble of prompts performs as well as when multiple hypernyms are given. This reflects the fact that the ensemble of prompts does induce BERT to predict hypernyms starting with a vowel, such as "animal" and "instrument". It is simply that the

Experiment	P@1	MRR	WN P@1	WN P@5
A(n) x is a [MASK].	30.3	44.4	36.2	22.0
A(n) x is a type of [MASK].	16.2	33.7	21.7	21.4
a [MASK], such as a(n) x	48.3	50.4	54.9	25.5
My favorite [MASK] is a(n) x.	35.7	44.0	50.8	23.8
[Handwritten context]. A(n) x is a [MASK].	42.5	53.0	45.1	18.6
[SemCor context]. A(n) x is a [MASK].	29.7	38.2	33.0	15.9
A(n) x is a [MASK]. So is a(n) x'.	51.7	63.0	54.1	23.8
A(n) x is a [MASK]. So is a(n) x'. (FT)	41.2	52.2	45.7	21.1
My favorite [MASK] is either a(n) x or a(n) x'. (FT)	38.5	47.1	55.9	24.7
Ensemble	29.8	44.9	38.6	26.9

Table 1: Diagnostic test performance (Battig dataset) when including the hyponym x in the top k. "WN" denotes WordNet, and "FT" denotes FastText.

Experiment	P@1	MRR	WN P@1	WN P@5
A(n) x is a [MASK].	38.7	49.7	46.6	22.9
A(n) x is a type of [MASK].	31.4	43.4	43.0	22.7
a [MASK], such as a(n) x	50.3	52.7	57.6	26.3
My favorite [MASK] is a(n) x.	35.7	44.1	50.8	23.9
[Handwritten context]. A(n) x is a [MASK].	44.8	54.7	48.2	19.0
[SemCor context]. x is a [MASK].	33.6	41.5	40.3	17.7
A(n) x is a [MASK]. So is a(n) x'.	57.4	66.7	61.5	24.2
A(n) x is a [MASK]. So is a(n) x'. (FT)	47.2	56.1	53.1	22.0
My favorite [MASK] is either a(n) x or a(n) x'. (FT)	38.6	47.4	56.5	24.9
Ensemble	38.2	50.3	53.0	30.0

Table 2: Diagnostic test performance (Battig dataset) when excluding the hyponym x from the top k. "WN" denotes WordNet, and "FT" denotes FastText.

hypernym predicted given this prompt is (in 15% of examples) correct, but not the canonical hypernym.

4 Comparison with other Hypernym Discovery Models

Although the goal of this paper is to learn more about BERT, we also compare BERT to other models used for hypernym discovery, the task of predicting a word's hypernym. To do so, we evaluate on the test split of the Semeval 2018 Task 9 (Camacho-Collados et al., 2018) dataset, which consists of 1500 words, each with a list of valid hypernyms.

This dataset is not very compatible with our approach. The hypernyms listed for each hyponym do not necessarily fit into one token. Moreover, the hyponyms in the dataset are abstract nouns. Many are also not necessarily common knowledge, being instead specific individuals, places, or technical terms. In contrast, the category norms dataset was designed to include only hyponyms that almost anyone would know.

As in the prior experiments, we use the bert-base-uncased pretrained BERT model from Huggingface (Wolf et al., 2020). We test the prompts "x is a type of [MASK]" and "My favorite [MASK] is x", to account for hypernyms starting with either consonants or vowels. We also test the high-performing "such-as" prompt "a [MASK], such as x". We predict the top 15 hypernyms for each hypernym and evaluate using the following metrics: mean average precision (MAP), Mean Reciprocal Rank (MRR), and precision at k (P@k). Mean average precision can be defined as the average over k of P@k; we compute this average for $k = 1, \dots, 15$. Table 3 shows the results for our BERT model as compared to the highest and lowest scoring (by MAP) supervised and unsupervised models (results from (Camacho-Collados et al., 2018)); our models are "BERT (such as)", "BERT (favorite)", and "BERT (type of)".

For all prompts, BERT outperforms all of the unsupervised models on all metrics, as well as the worst-performing supervised model. The "such-as" prompt even manages to outperform the supervised

Model	S/US	MAP	MRR	P@5
BERT (such as)	US	20.17	12.65	10.49
CRIM_r1	S	19.78	36.10	19.03
SJTU BCMI	S	5.77	10.56	5.96
BERT (favorite)	US	9.17	13.95	7.91
BERT (type of)	US	7.37	22.30	8.81
Team 13	US	2.77	6.07	2.72
Apollo_r1	US	1.40	3.51	1.33

Table 3: Model performance on Semeval 2018 Task 9: Hypernym Discovery (Semeval 2018 Task 9 Dataset). S/US = Supervised or Unsupervised

models with respect to MAP, although its MRR and P@5 put it squarely in the middle of the pack. This is surprising, considering that BERT makes no use of the closed vocabulary of hypernyms that this task has, and other models took advantage of. In fact, with the "such-as" prompt, BERT could not possibly predict many of the hypernyms in this task, including hypernyms that start with vowels, and hypernyms that are represented as multiple tokens. However, its performance is still respectable.

5 Qualitative analysis - does BERT understand hypernymy?

Having quantitatively analyzed BERT's outputs on various prompts, we now turn to qualitatively analyze them. The immediate results—that the quality of BERT's responses depends heavily on the prompt, and none of the prompts are perfect—are not terribly surprising, and follow from those of other works on prompting. However, a closer look reveals that while no prompt was successful at recovering the hypernyms we desired, *certain* prompts were able to most often produce reasonable hypernyms, even if they were not exactly the desired hypernyms.

Past works (Ravichander et al., 2020) have suggested that BERT's answers to hypernym queries show a lack of knowledge of hypernymy. Some of our prompts display the same behavior. For example, the "type-of" prompt yields "wine", "coffee", and "grape" for the hypernyms of the color "mauve". BERT guesses items that may be mauve in color, but are certainly not hypernyms of mauve; this prompt suggests that BERT does not understand the hypernymy relation.

However, the performance of the "such-as" prompt tells a different story. It is, in fact, more accurate than the numbers suggest—it predicts in the Battig dataset, for example, "felony" to be the hypernym of "embezzlement", which is marked wrong, but is in fact correct. Follow-up manual analysis of 100 (out of 419) incorrect predictions from the Battig dataset using this prompt showed that 35% of answers marked as incorrect were actually correct, suggesting a true P@1 of 66.4%. This is also true of the "favorite" prompt, in which the same analysis suggested a true P@1 of 76%, higher than the "such-as" prompt's.

Despite this seemingly-good performance, a closer look at the incorrect answers provided by BERT in response to these prompts complicates the notion that BERT understands hypernymy. When using the "such-as" prompt, a pattern of errors appears with respect to city names: an oft-guessed hypernym is "few". While this makes sense in the context of the prompt—consider a phrase such as "most cities, except for a [few, such as Minneapolis]"—this is not a valid hypernym. However, it is a convenient failure case that BERT can predict for almost any hyponym; it appears in the top 15 predictions for 438 of 1500 entries in the SemEval dataset.

A similar pattern appears with the "favorite prompt". On the Battig dataset, BERT sometimes yields hypernyms that are too generic—for example, as the hypernyms of medicine, it yields "thing", "subject", and "topic". Worse, on the SemEval dataset BERT predicts "word" as the top hypernym for 305 of the 1500 words in the dataset; while "word" is a grammatically and logically correct answer to the prompt, it does not show any understanding of hypernymy. Furthermore, the next 4 top predictions by BERT cover another 427 of the 1500 entries, suggesting that BERT is unable to provide a meaningful hypernym for each word. In contrast, the top hypernym on the Semeval dataset for the "type-of" prompt is assigned to only 37 of 1500 words. So, the "type-of" prompt does not guess the same hypernym for many words.

In light of these phenomena, it is difficult to say whether either of these two high-performing prompts has actually elicited knowledge of hypernymy from BERT. The "such-as" prompt performs well, and if not for vowel / consonant issues, might perform even better. Unfortunately, since the prompt does not specifically ask for the hypernym of x, its incorrect answers leave unclear whether BERT is failing to systematically understand hypernymy, or is simply performing general language modeling when we would prefer that it extract a hypernym.

The "favorite" prompt also performs well, and even its failures are often correct, albeit too-generic, hypernyms. The prompt also implies that "`[MASK]`" is the hypernym of x. Moreover, because some of the hypernyms (e.g. "crime") are rather unlikely to have appeared in BERT's training data in the context of the prompt (i.e. because "My favorite crime is..." is not a common phrase), it seems reasonable to attribute some of BERT's performance to genuine learning of hypernymy, rather than simple memorization of training data. However, this is not true of all hypernyms: it is possible that BERT's use of generic hypernyms like "thing" stems from a memorization of frequent constructions like "my favorite thing" in the training data.

6 Conclusion

In this paper, we conduct a thorough investigation of BERT's knowledge of hypernymy via prompting. Inspired by recent work on prompting, we use a wide variety of prompts to search for knowledge of hypernymy. We find that BERT does have some knowledge of hypernymy, and performs better than other unsupervised models for hypernym discovery. Furthermore, two prompts, "`[MASK]`, such as x" and "My favorite `[MASK]` is x.", often elicit correct hypernyms from BERT. However, even with these prompts, BERT occasionally fails, producing either non-hypernyms or extremely general hypernyms. Moreover, this prompting methodology does not allow us to distinguish between understanding of hypernymy and memorization of hypernyms from training data. Thus, we conclude that while some of the prompts elicited many correct hypernyms, we cannot claim that BERT has fully understood hypernymy.

Acknowledgments

We thank anonymous reviewers for their valuable comments and suggestions for improvement.

This work was supported by the GAČ NEUREM3 grant (Neural Representations in Multi-modal and Multi-lingual Modelling, 19-26934X).

Computational resources were supplied by the project "e-Infrastruktura CZ" (e-INFRA CZ ID:90140) supported by the Ministry of Education, Youth and Sports of the Czech Republic.

References

William F. Battig and William E. Montague. 1969. Category norms of verbal items in 56 categories a replication and extension of the connecticut category norms. *Journal of Experimental Psychology*, 80(3p2):1.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Jose Camacho-Collados, Claudio Delli Bovi, Luis Espinosa-Anke, Sergio Oramas, Tommaso Pasini, Enrico Santus, Vered Shwartz, Roberto Navigli, and Horacio Saggion. 2018. SemEval-2018 task 9: Hypernym discovery. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 712–724, New Orleans, Louisiana. Association for Computational Linguistics.

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 (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia. Association for Computational Linguistics.

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

Allyson Ettinger. 2020. What BERT Is Not: Lessons from a New Suite of Psycholinguistic Diagnostics for Language Models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. Bradford Books.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *COLING 1992 Volume 2: The 14th International Conference on Computational Linguistics*.

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 and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2733–2743, Hong Kong, China. Association for Computational Linguistics.

Zhengbao Jiang, Frank F. Xu, J. Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Helen Langone, Benjamin R. Haskell, and George A. Miller. 2004. Annotating WordNet. In *Proceedings of the Workshop Frontiers in Corpus Annotation at HLT-NAACL 2004*, pages 63–69, Boston, Massachusetts, USA. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Alec Radford, Jeff Wu, R. Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Abhilasha Ravichander, Eduard Hovy, Kaheer Suleman, Adam Trischler, and Jackie Chi Kit Cheung. 2020. On the systematicity of probing contextualized word representations: The case of hypernymy in BERT. In *Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics*, pages 88–102, Barcelona, Spain (Online). Association for Computational Linguistics.

Laria Reynolds and Kyle McDonell. 2021. *Prompt Programming for Large Language Models: Beyond the Few-Shot Paradigm*. Association for Computing Machinery, New York, NY, USA.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in bertology: What we know about how bert works.

Timo Schick and Hinrich Schütze. 2020. Rare words: A major problem for contextualized representation and how to fix it by attentive mimicking. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV au2, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts.

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

Ivan Vulić, Edoardo Maria Ponti, Robert Litschko, Goran Glavaš, and Anna Korhonen. 2020. Probing pretrained language models for lexical semantics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7222–7240, Online. Association for Computational Linguistics.

Gregor Wiedemann, Steffen Remus, Avi Chawla, and Chris Biemann. 2019. Does bert make any sense? interpretable word sense disambiguation with contextualized embeddings.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

David Yenicelik, Florian Schmidt, and Yannic Kilcher. 2020. How does BERT capture semantics? a closer look at polysemous words. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 156–162, Online. Association for Computational Linguistics.