Language Models as Knowledge Bases?

Fabio Petroni[1] **Tim Rocktäschel**[1,2] **Patrick Lewis**[1,2] **Anton Bakhtin**[1]
Yuxiang Wu[1,2] **Alexander H. Miller**[1] **Sebastian Riedel**[1,2]
[1]Facebook AI Research
[2]University College London
{fabiopetroni, rockt, plewis, yolo, yuxiangwu, ahm, sriedel}@fb.com

Abstract

Recent progress in pretraining language models on large textual corpora led to a surge of improvements for downstream NLP tasks. Whilst learning linguistic knowledge, these models may also be storing relational knowledge present in the training data, and may be able to answer queries structured as "fill-in-the-blank" cloze statements. Language models have many advantages over structured knowledge bases: they require no schema engineering, allow practitioners to query about an open class of relations, are easy to extend to more data, and require no human supervision to train. We present an in-depth analysis of the relational knowledge already present (without fine-tuning) in a wide range of state-of-the-art pretrained language models. We find that (i) without fine-tuning, BERT contains relational knowledge competitive with traditional NLP methods that have some access to oracle knowledge, (ii) BERT also does remarkably well on open-domain question answering against a supervised baseline, and (iii) certain types of factual knowledge are learned much more readily than others by standard language model pretraining approaches. The surprisingly strong ability of these models to recall factual knowledge without any fine-tuning demonstrates their potential as unsupervised open-domain QA systems. The code to reproduce our analysis is available at https://github.com/facebookresearch/LAMA.

1 Introduction

Recently, pretrained high-capacity language models such as ELMo (Peters et al., 2018a) and BERT (Devlin et al., 2018a) have become increasingly important in NLP. They are optimised to either predict the next word in a sequence or some masked word anywhere in a given sequence (*e.g.* "Dante was born in [MASK] in the year 1265."). The parameters of these models appear to store

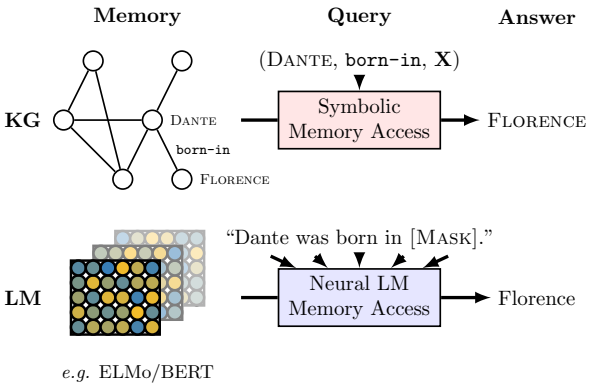

Figure 1: Querying knowledge bases (KB) and language models (LM) for factual knowledge.

vast amounts of linguistic knowledge (Peters et al., 2018b; Goldberg, 2019; Tenney et al., 2019) useful for downstream tasks. This knowledge is usually accessed either by conditioning on latent context representations produced by the original model or by using the original model weights to initialize a task-specific model which is then further fine-tuned. This type of knowledge transfer is crucial for current state-of-the-art results on a wide range of tasks.

In contrast, knowledge bases are effective solutions for accessing annotated gold-standard relational data by enabling queries such as (DANTE, born-in, **X**). However, in practice we often need to *extract* relational data from text or other modalities to populate these knowledge bases. This requires complex NLP pipelines involving entity extraction, coreference resolution, entity linking and relation extraction (Surdeanu and Ji, 2014)—components that often need supervised data and fixed schemas. Moreover, errors can easily propagate and accumulate throughout the pipeline. Instead, we could attempt to query neural language models for relational data by asking them to fill in masked tokens in sequences like "Dante was born

in [MASK]", as illustrated in Figure 1. In this setting, language models come with various attractive properties: they require no schema engineering, do not need human annotations, and they support an open set of queries.

Given the above qualities of language models as potential representations of relational knowledge, we are interested in the relational knowledge already present in pretrained *off-the-shelf* language models such as ELMo and BERT. How much relational knowledge do they store? How does this differ for different types of knowledge such as facts about entities, common sense, and general question answering? How does their performance without fine-tuning compare to symbolic knowledge bases automatically extracted from text? Beyond gathering a better general understanding of these models, we believe that answers to these questions can help us design better unsupervised knowledge representations that could transfer factual and commonsense knowledge reliably to downstream tasks such as commonsense (visual) question answering (Zellers et al., 2018; Talmor et al., 2019) or reinforcement learning (Branavan et al., 2011; Chevalier-Boisvert et al., 2018; Bahdanau et al., 2019; Luketina et al., 2019).

For the purpose of answering the above questions we introduce the *LAMA* (LAnguage Model Analysis) probe, consisting of a set of knowledge sources, each comprised of a set of facts. We define that a pretrained language model *knows* a fact (SUBJECT, `relation`, OBJECT) such as (DANTE, `born-in`, FLORENCE) if it can successfully predict masked objects in cloze sentences such as "Dante was born in ____" expressing that fact. We test for a variety of types of knowledge: relations between entities stored in Wikidata, common sense relations between concepts from ConceptNet, and knowledge necessary to answer natural language questions in SQuAD. In the latter case we manually map a subset of SQuAD questions to cloze sentences.

Our investigation reveals that (i) the largest BERT model from Devlin et al. (2018b) (BERT-large) captures (accurate) relational knowledge comparable to that of a knowledge base extracted with an off-the-shelf relation extractor and an oracle-based entity linker from a corpus known to express the relevant knowledge, (ii) factual knowledge can be recovered surprisingly well from pretrained language models, however, for some relations (particularly N-to-M relations) performance is very poor, (iii) BERT-large consistently outperforms other language models in recovering factual and commonsense knowledge while at the same time being more robust to the phrasing of a query, and (iv) BERT-large achieves remarkable results for open-domain QA, reaching 57.1% precision@10 compared to 63.5% of a knowledge base constructed using a task-specific supervised relation extraction system.

2 Background

In this section we provide background on language models. Statistics for the models that we include in our investigation are summarized in Table 1.

2.1 Unidirectional Language Models

Given an input sequence of tokens $\mathbf{w} = [w_1, w_2, \ldots, w_N]$, unidirectional language models commonly assign a probability $p(\mathbf{w})$ to the sequence by factorizing it as follows

$$p(\mathbf{w}) = \prod_t p(w_t \,|\, w_{t-1}, \ldots, w_1). \qquad (1)$$

A common way to estimate this probability is using neural language models (Mikolov and Zweig, 2012; Melis et al., 2017; Bengio et al., 2003) with

$$p(w_t \,|\, w_{t-1}, \ldots, w_1) = \text{softmax}(\mathbf{W}\mathbf{h}_t + \mathbf{b}) \qquad (2)$$

where $\mathbf{h}_t \in \mathbb{R}^k$ is the output vector of a neural network at position t and $\mathbf{W} \in \mathbb{R}^{|\mathcal{V}| \times k}$ is a learned parameter matrix that maps $\mathbf{h}_t$ to unnormalized scores for every word in the vocabulary $\mathcal{V}$. Various neural language models then mainly differ in how they compute $\mathbf{h}_t$ given the word history, *e.g.*, by using a multi-layer perceptron (Bengio et al., 2003; Mikolov and Zweig, 2012), convolutional layers (Dauphin et al., 2017), recurrent neural networks (Zaremba et al., 2014; Merity et al., 2016; Melis et al., 2017) or self-attention mechanisms (Radford et al., 2018; Dai et al., 2019; Radford et al., 2019).

fairseq-fconv: Instead of commonly used recurrent neural networks, Dauphin et al. (2017) use multiple layers of gated convolutions. We use the pretrained model in the fairseq[1] library in our study. It has been trained on the WikiText-103 corpus introduced by Merity et al. (2016).

[1] https://github.com/pytorch/fairseq

Model	Base Model	#Parameters	Training Corpus	Corpus Size
fairseq-fconv (Dauphin et al., 2017)	ConvNet	324M	WikiText-103	103M Words
Transformer-XL (large) (Dai et al., 2019)	Transformer	257M	WikiText-103	103M Words
ELMo (original) (Peters et al., 2018a)	BiLSTM	93.6M	Google Billion Word	800M Words
ELMo 5.5B (Peters et al., 2018a)	BiLSTM	93.6M	Wikipedia (en) & WMT 2008-2012	5.5B Words
BERT (base) (Devlin et al., 2018a)	Transformer	110M	Wikipedia (en) & BookCorpus	3.3B Words
BERT (large) (Devlin et al., 2018a)	Transformer	340M	Wikipedia (en) & BookCorpus	3.3B Words

Table 1: Language models considered in this study.

Transformer-XL: Dai et al. (2019) introduce a large-scale language model based on the Transformer (Vaswani et al., 2017). Transformer-XL can take into account a longer history by caching previous outputs and by using relative instead of absolute positional encoding. It achieves a test perplexity of 18.3 on the WikiText-103 corpus.

2.2 Bidirectional "Language Models"[2]

So far, we have looked at language models that predict the next word given a history of words. However, in many downstream applications we mostly care about having access to contextual representations of words, *i.e.*, word representations that are a function of the entire context of a unit of text such as a sentence or paragraph, and not only conditioned on previous words. Formally, given an input sequence $\mathbf{w} = [w_1, w_2, \ldots, w_N]$ and a position $1 \leq i \leq N$, we want to estimate $p(w_i) = p(w_i \mid w_1, \ldots, w_{i-1}, w_{i+1}, \ldots, w_N)$ using the left and right context of that word.

ELMo: To estimate this probability, Peters et al. (2018a) propose running a forward and backward LSTM (Hochreiter and Schmidhuber, 1997), resulting in $\overrightarrow{\mathbf{h}}_i$ and $\overleftarrow{\mathbf{h}}_i$ which consequently are used to calculate a forward and backward language model log-likelihood. Their model, ELMo, uses multiple layers of LSTMs and it has been pretrained on the Google Billion Word dataset. Another version of the model, ELMo 5.5B, has been trained on the English Wikipedia and monolingual news crawl data from WMT 2008-2012.

BERT: Instead of a standard language model objective, Devlin et al. (2018a) propose to sample positions in the input sequence randomly and to learn to fill the word at the masked position. To this end, they employ a Transformer architecture and train it on the BookCorpus (Zhu et al., 2015) as well as a crawl of English Wikipedia. In addition to this pseudo language model objective, they use an auxiliary binary classification objective to predict whether a particular sentence follows the given sequence of words.

[2]Contextual representation models (Tenney et al., 2019) might be a better name, but we keep calling them language models for simplicity.

3 Related Work

Many studies have investigated pretrained word representations, sentence representations, and language models. Existing work focuses on understanding linguistic and semantic properties of word representations or how well pretrained sentence representations and language models transfer linguistic knowledge to downstream tasks. In contrast, our investigation seeks to answer to what extent pretrained language models store factual and commonsense knowledge by comparing them with symbolic knowledge bases populated by traditional relation extraction approaches.

Baroni et al. (2014) present a systematic comparative analysis between neural word representation methods and more traditional count-based distributional semantic methods on lexical semantics tasks like semantic relatedness and concept categorization. They find that neural word representations outperform count-based distributional methods on the majority of the considered tasks. Hill et al. (2015) investigate to what degree word representations capture semantic meaning as measured by similarity between word pairs.

Marvin and Linzen (2018) assess the grammaticality of pretrained language models. Their dataset consists of sentence pairs with a grammatical and an ungrammatical sentence. While a good language model should assign higher probability to the grammatical sentence, they find that LSTMs do not learn syntax well.

Another line of work investigates the ability of pretrained sentence and language models to transfer knowledge to downstream natural language understanding tasks (Wang et al., 2018). While such an analysis sheds light on the transfer-learning

abilities of pretrained models for understanding short pieces of text, it provides little insight into whether these models can compete with traditional approaches to representing knowledge like symbolic knowledge bases.

More recently, McCoy et al. (2019) found that for natural language inference, a model based on BERT learns to rely heavily on fallible syntactic heuristics instead of a deeper understanding of the natural language input. Peters et al. (2018b) found that lower layers in ELMo specialize on local syntactic relationships, while higher layers can learn to model long-range relationships. Similarly, Goldberg (2019) found that BERT captures English syntactic phenomena remarkably well. Tenney et al. (2019) investigate to what extent language models encode sentence structure for different syntactic and semantic phenomena and found that they excel for the former but only provide small improvements for tasks that fall into the latter category. While this provides insights into the linguistic knowledge of language models, it does not provide insights into their factual and commonsense knowledge.

Radford et al. (2018) introduce a pretrained language model based on the Transformer which they termed generative pretraining (GPTv1). The first version of GPT (Radford et al., 2018) has been trained on the Book Corpus (Zhu et al., 2015) containing 7000 books. The closest to our investigation is the work by Radford et al. (2019) which introduces GPTv2 and investigates how well their language model does zero-shot transfer to a range of downstream tasks. They find that GPTv2 achieves an F_1 of 55 for answering questions in CoQA (Reddy et al., 2018) and 4.1% accuracy on the Natural Questions dataset (Kwiatkowski et al., 2019), in both cases without making use of annotated question-answer pairs or an information retrieval step. While these results are encouraging and hint at the ability of very large pretrained language models to memorize factual knowledge, the large GPTv2 model has not been made public and the publicly available small version achieves less than 1% on Natural Questions (5.3 times worse than the large model). Thus, we decided to not include GPTv2 in our study. Similarly, we do not include GPTv1 in this study as it uses a limited lower-cased vocabulary, making it incompatible to the way we assess the other language models.

4 The LAMA Probe

We introduce the *LAMA* (LAnguage Model Analysis) probe to test the factual and commonsense knowledge in language models. It provides a set of knowledge sources which are composed of a corpus of facts. Facts are either subject-relation-object triples or question-answer pairs. Each fact is converted into a cloze statement which is used to query the language model for a missing token. We evaluate each model based on how highly it ranks the ground truth token against every other word in a fixed candidate vocabulary. This is similar to ranking-based metrics from the knowledge base completion literature (Bordes et al., 2013; Nickel et al., 2016). Our assumption is that models which rank ground truth tokens high for these cloze statements have more factual knowledge. We discuss each step in detail next and provide considerations on the probe below.

4.1 Knowledge Sources

To assess the different language models in Section 2, we cover a variety of sources of factual and commonsense knowledge. For each source, we describe the origin of fact triples (or question-answer pairs), how we transform them into cloze templates, and to what extent aligned texts exist in Wikipedia that are known to express a particular fact. We use the latter information in supervised baselines that extract knowledge representations directly from the aligned text.

4.1.1 Google-RE

The Google-RE corpus[3] contains ~60K facts manually extracted from Wikipedia. It covers five relations but we consider only three of them, namely "place of birth", "date of birth" and "place of death". We exclude the other two because they contain mainly multi-tokens objects that are not supported in our evaluation. We manually define a template for each considered relation, *e.g.*, "[S] was born in [O]" for "place of birth". Each fact in the Google-RE dataset is, by design, manually aligned to a short piece of Wikipedia text supporting it.

4.1.2 T-REx

The T-REx knowledge source is a subset of Wikidata triples. It is derived from the T-REx

[3] https://code.google.com/archive/p/relation-extraction-corpus/

dataset (Elsahar et al., 2018) and is much larger than Google-RE with a broader set of relations. We consider 41 Wikidata relations and subsample at most 1000 facts per relation. As with the Google-RE corpus, we manually define a template for each relation (see Table 3 for some examples). In contrast to the Google-RE knowledge source, T-REx facts were automatically aligned to Wikipedia and hence this alignment can be noisy. However, Elsahar et al. (2018) report an accuracy of 97.8% for the alignment technique over a test set.

4.1.3 ConceptNet

ConceptNet (Speer and Havasi, 2012) is a multilingual knowledge base, initially built on top of Open Mind Common Sense (OMCS) sentences. OMCS represents commonsense relationships between words and/or phrases. We consider facts from the English part of ConceptNet that have single-token objects covering 16 relations. For these ConceptNet triples, we find the OMCS sentence that contains both the subject and the object. We then mask the object within the sentence and use the sentence as template for querying language models. If there are several sentences for a triple, we pick one at random. Note that for this knowledge source there is no explicit alignment of facts to Wikipedia sentences.

4.1.4 SQuAD

SQuAD (Rajpurkar et al., 2016) is a popular question answering dataset. We select a subset of 305 context-insensitive questions from the SQuAD development set with single token answers. We manually create cloze-style questions from these questions, *e.g.*, rewriting "Who developed the theory of relativity?" as "The theory of relativity was developed by ____". For each question and answer pair, we know that the corresponding fact is expressed in Wikipedia since this is how SQuAD was created.

4.2 Models

We consider the following pretrained case-sensitive language models in our study (see Table 1): fairseq-fconv (*Fs*), Transformer-XL large (*Txl*), ELMo original (*Eb*), ELMo 5.5B (*E5B*), BERT-base (*Bb*) and BERT-large (*Bl*). We use the natural way of generating tokens for each model by following the definition of the training objective function.

Assume we want to compute the generation for the token at position t. For unidirectional language models, we use the network output ($\mathbf{h}_{t-1}$) just before the token to produce the output layer softmax. For ELMo we consider the output just before ($\overrightarrow{\mathbf{h}}_{t-1}$) for the forward direction and just after ($\overleftarrow{\mathbf{h}}_{t+1}$) for the backward direction. Following the loss definition in (Peters et al., 2018a), we average forward and backward probabilities from the corresponding softmax layers. For BERT, we mask the token at position t, and we feed the output vector corresponding to the masked token ($\mathbf{h}_t$) into the softmax layer. To allow a fair comparison, we let models generate over a unified vocabulary, which is the intersection of the vocabularies for all considered models (~21K case-sensitive tokens).

4.3 Baselines

To compare language models to canonical ways of using off-the-shelf systems for extracting symbolic knowledge and answering questions, we consider the following baselines.

Freq: For a subject and relation pair, this baseline ranks words based on how frequently they appear as objects for the given relation in the test data. It indicates the upper bound performance of a model that always predicts the same objects for a particular relation.

RE: For the relation-based knowledge sources, we consider the pretrained Relation Extraction (RE) model of Sorokin and Gurevych (2017). This model was trained on a subcorpus of Wikipedia annotated with Wikidata relations. It extracts relation triples from a given sentence using an LSTM-based encoder and an attention mechanism. Based on the alignment information from the knowledge sources, we provide the relation extractor with the sentences known to express the test facts. Using these datasets, RE constructs a knowledge graph of triples. At test time, we query this graph by finding the subject entity and then rank all objects in the correct relation based on the confidence scores returned by RE. We consider two versions of this procedure that differ in how the entity linking is implemented: $\mathbf{RE}_n$ makes use of a naïve entity linking solution based on exact string matching, while $\mathbf{RE}_o$ uses an oracle for entity linking in addition to string matching. In other words, assume we query for the object o of a test subject-relation fact (s, r, o) expressed in a sentence x. If RE has extracted any triple (s', r, o') from that sen-

tence x, s' will be linked to s and o' to o. In practice, this means RE can return the correct solution o if *any* relation instance of the right type was extracted from x, regardless of whether it has a wrong subject or object.

DrQA: Chen et al. (2017) introduce DrQA, a popular system for open-domain question answering. DrQA predicts answers to natural language questions using a two step pipeline. First, a TF/IDF information retrieval step is used to find relevant articles from a large store of documents (e.g. Wikipedia). On the retrieved top k articles, a neural reading comprehension model then extracts answers. To avoid giving the language models a competitive advantage, we constrain the predictions of DrQA to single-token answers.

4.4 Metrics

We consider rank-based metrics and compute results per relation along with mean values across all relations. To account for multiple valid objects for a subject-relation pair (*i.e.*, for N-M relations), we follow Bordes et al. (2013) and remove from the candidates when ranking at test time all other valid objects in the training data other than the one we test. We use the mean precision at k ($P@k$). For a given fact, this value is 1 if the object is ranked among the top k results, and 0 otherwise.

4.5 Considerations

There are several important design decisions we made when creating the LAMA probe. Below we give more detailed justifications for these decisions.

Manually Defined Templates For each relation we manually define a template that queries for the object slot in that relation. One can expect that the choice of templates has an impact on the results, and this is indeed the case: for some relations we find both worse and better ways to query for the same information (with respect to a given model) by using an alternate template. We argue that this means we are measuring a *lower* bound for what language models know. We make this argument by analogy with traditional knowledge bases: they only have a *single* way of querying knowledge for a specific relation, namely by using the relation id of that relation, and this way is used to measure their accuracy. For example, if the relation ID is `works-For` and the user asks for `is-working-for`, the accuracy of the KG would be 0.

Single Token We only consider single token objects as our prediction targets. The reason we include this limitation is that multi-token decoding adds a number of additional tuneable parameters (beam size, candidate scoring weights, length normalization, n-gram repetition penalties, etc.) that obscure the knowledge we are trying to measure. Moreover, well-calibrated multi-token generation is still an active research area, particularly for bidirectional models (see *e.g.* Welleck et al. (2019)).

Object Slots We choose to *only* query object slots in triples, as opposed to subject or relation slots. By including reverse relations (e.g. `contains` and `contained-by`) we can also query subject slots. We do not query relation slots for two reasons. First, surface form realisations of relations will span several tokens, and as we discussed above, this poses a technical challenge that is not in the scope of this work. Second, even if we could easily predict multi-token phrases, relations can generally be expressed with many different wordings, making it unclear what the gold standard pattern for a relation should be, and how to measure accuracy in this context.

Intersection of Vocabularies The models that we considered are trained with different vocabularies. For instance, ELMo uses a list of ~800K tokens while BERT considers only ~30K tokens. The size of the vocabulary can influence the performance of a model for the LAMA probe. Specifically, the larger the vocabulary the harder it would be to rank the gold token at the top. For this reason we considered a common vocabulary of ~21K case-sensitive tokens that are obtained from the intersection of the vocabularies for all considered models. To allow a fair comparison, we let every model rank only tokens in this joint vocabulary.

5 Results

We summarize the main results in Table 2, which shows the mean precision at one (P@1) for the different models across the set of corpora considered. In the remainder of this section, we discuss the results for each corpus in detail.

Google-RE We query the LMs using a standard cloze template for each relation. The base and large versions of BERT both outperform all other models by a substantial margin. Furthermore, they

Corpus	Relation	Statistics		Baselines		KB		LM					
		#Facts	#Rel	Freq	DrQA	RE_n	RE_o	Fs	Txl	Eb	E5B	Bb	Bl
Google-RE	birth-place	2937	1	4.6	-	3.5	13.8	4.4	2.7	5.5	7.5	14.9	**16.1**
	birth-date	1825	1	1.9	-	0.0	**1.9**	0.3	1.1	0.1	0.1	1.5	1.4
	death-place	765	1	6.8	-	0.1	7.2	3.0	0.9	0.3	1.3	13.1	**14.0**
	Total	5527	3	4.4	-	1.2	7.6	2.6	1.6	2.0	3.0	9.8	**10.5**
T-REx	1-1	937	2	1.78	-	0.6	10.0	17.0	36.5	10.1	13.1	68.0	**74.5**
	N-1	20006	23	23.85	-	5.4	**33.8**	6.1	18.0	3.6	6.5	32.4	34.2
	N-*M*	13096	16	21.95	-	7.7	**36.7**	12.0	16.5	5.7	7.4	24.7	24.3
	Total	34039	41	22.03	-	6.1	**33.8**	8.9	18.3	4.7	7.1	31.1	32.3
ConceptNet	Total	11458	16	4.8	-	-	-	3.6	5.7	6.1	6.2	15.6	**19.2**
SQuAD	Total	305	-	-	**37.5**	-	-	3.6	3.9	1.6	4.3	14.1	17.4

Table 2: Mean precision at one (P@1) for a frequency baseline (Freq), DrQA, a relation extraction with naïve entity linking (RE_n), oracle entity linking (RE_o), fairseq-fconv (Fs), Transformer-XL large (Txl), ELMo original (Eb), ELMo 5.5B (E5B), BERT-base (Bb) and BERT-large (Bl) across the set of evaluation corpora.

obtain a 2.2 and 2.9 respective average accuracy improvement over the oracle-based RE baseline. This is particularly surprising given that with the gold-aligned Google-RE source we know for certain that the oracle RE baseline has seen at least one sentence expressing each test fact. Moreover, the RE baseline was given substantial help through an entity linking oracle.

It is worth pointing out that while BERT-large does better, this does not mean it does so for the right reasons. Although the aligned Google-RE sentences are likely in its training set (as they are part of Wikipedia and BERT has been trained on Wikipedia), it might not "understand" them to produce these results. Instead, it could have learned associations of objects with subjects from co-occurrence patterns.

T-REx The knowledge source derived from Google-RE contains relatively few facts and only three relations. Hence, we perform experiments on the larger set of facts and relations in T-REx. We find that results are generally consistent with Google-RE. Again, the performance of BERT in retrieving factual knowledge are close to the performance obtained by automatically building a knowledge base with an off-the-shelf relation extraction system and oracle-based entity linking. Broken down by relation type, the performance of BERT is very high for 1-to-1 relations (*e.g.*, *capital of*) and low for N-to-M relations.

Note that a downstream model could learn to make use of knowledge in the output representations of a language model even if the correct answer is not ranked first but high enough (*i.e.* a hint

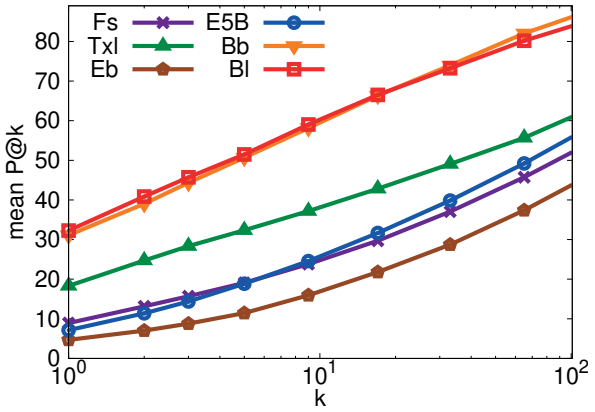

Figure 2: Mean P@k curve for T-REx varying k. Base-10 log scale for X axis.

about the correct answer can be extracted from the output representation). Figure 2 shows the mean P@k curves for the considered models. For BERT, the correct object is ranked among the top ten in around 60% of the cases and among the top 100 in 80% of the cases.

To further investigate why BERT achieves such strong results, we compute the Pearson correlation coefficient between the *P@1* and a set of metrics that we report in Figure 3. We notice, for instance, that the number of times an object is mentioned in the training data positively correlates with performance while the same is not true for the subject of a relation. Furthermore, the log probability of a prediction is strongly positively correlated with P@1. Thus, when BERT has a high confidence in its prediction, it is often correct. Performance is also positively correlated with the cosine similarity between subject and object vectors, and

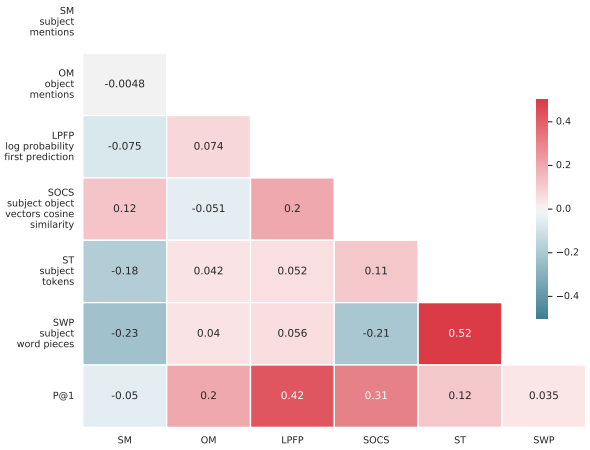

Figure 3: Pearson correlation coefficient for the P@1 of the BERT-large model on T-REx and a set of metrics: SM and OM refer to the number of times a subject and an object are mentioned in the BERT training corpus[4]respectively; LPFP is the log probability score associated with the first prediction; SOCS is the cosine similarity between subject and object vectors (we use spaCy[5]); ST and SWP are the number of tokens in the subject with a standard tokenization and the BERT WordPiece tokenization respectively.

slightly with the number of tokens in the subject.

Table 3 shows randomly picked examples for the generation of BERT-large for cloze template queries. We find that BERT-large generally predicts objects of the correct type, even when the predicted object itself is not correct.

To understand how the performance of a pretrained language model varies with different ways of querying for a particular fact, we analyze a maximum of 100 random facts per relation for which we randomly select 10 aligned sentences in Wikipedia from T-REx.[6] In each of the sentences, we mask the object of the fact, and ask the model to predict it. For several of our language models this also tests their ability to memorize and recall sentences from the training data since as the models have been trained on Wikipedia (see Table 1).

Figure 4 shows the average distribution of the rank for ten queries per fact. The two BERT models and ELMo 5.5B exhibit the lowest variability while ranking the correct object close to the top on average. Surprisingly, the performance of ELMo original is not far from BERT, even though this model did not see Wikipedia during training. Fairseq-fconv and Transformer-XL experience a higher variability in their predictions. Note that BERT and ELMo 5.5B have been trained on a larger portion of Wikipedia than fairseq-fconv and Transformer-XL and may have seen more sentences containing the test queries during training.

[5]The original training corpus is not available, we created our version using the same sources.

[5]https://spacy.io

[6]We exclude all facts with less than 10 alignments.

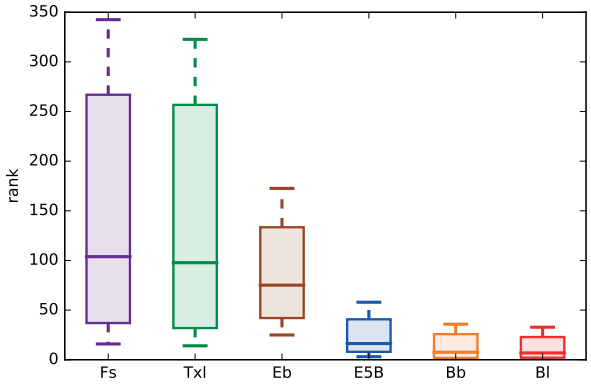

Figure 4: Average rank distribution for 10 different mentions of 100 random facts per relation in T-REx. ELMo 5.5B and both variants of BERT are least sensitive to the framing of the query but also are the most likely to have seen the query sentence during training.

ConceptNet The results on the ConceptNet corpus are in line with those reported for retrieving factual knowledge in Google-RE and T-REx. The BERT-large model consistently achieves the best performance, and it is able to retrieve commonsense knowledge at a similar level to factual knowledge. The lower half of Table 3 shows generations by BERT-large for randomly sampled examples. Some of the concepts generated by the language models are surprisingly reasonable in addition to being syntactically correct.

SQuAD Next we evaluate our system on open-domain cloze-style question answering and compare against the supervised DrQA model. Table 2 shows a performance gap between BERT-large and the DrQA open-domain QA system on our cloze SQuAD task. Again, note that the pretrained language model is completely unsupervised, it is not fine-tuned, and it has no access to a dedicated information retrieval system. Moreover, when comparing DrQA and BERT-large in terms of P@10, we find that gap is remarkably small (57.1 for BERT-large and 63.5 for DrQA).

6 Discussion and Conclusion

We presented a systematic analysis of the factual and commonsense knowledge in publicly available pretrained language models *as is* and found

	Relation	Query	Answer	Generation
T-Rex	P19	Francesco Bartolomeo Conti was born in ____.	Florence	Rome [-1.8] , **Florence [-1.8]** , Naples [-1.9] , Milan [-2.4] , Bologna [-2.5]
	P20	Adolphe Adam died in ____.	Paris	**Paris [-0.5]** , London [-3.5] , Vienna [-3.6] , Berlin [-3.8] , Brussels [-4.0]
	P279	English bulldog is a subclass of ____.	dog	dogs [-0.3] , breeds [-2.2] , **dog [-2.4]** , cattle [-4.3] , sheep [-4.5]
	P37	The official language of Mauritius is ____.	English	**English [-0.6]** , French [-0.9] , Arabic [-6.2] , Tamil [-6.7] , Malayalam [-7.0]
	P413	Patrick Oboya plays in ____ position.	midfielder	centre [-2.0] , center [-2.2] , **midfielder [-2.4]** , forward [-2.4] , midfield [-2.7]
	P138	Hamburg Airport is named after ____.	Hamburg	Hess [-7.0] , Hermann [-7.1] , Schmidt [-7.1] , **Hamburg [-7.5]** , Ludwig [-7.5]
	P364	The original language of Mon oncle Benjamin is ____.	French	**French [-0.2]** , Breton [-3.3] , English [-3.8] , Dutch [-4.2] , German [-4.9]
	P54	Dani Alves plays with ____ .	Barcelona	Santos [-2.4] , Porto [-2.5] , Sporting [-3.1] , Brazil [-3.3] , Portugal [-3.7]
	P106	Paul Toungui is a ____ by profession .	politician	lawyer [-1.1] , journalist [-2.4] , teacher [-2.7] , doctor [-3.0] , physician [-3.7]
	P527	Sodium sulfide consists of ____.	sodium	water [-1.2] , sulfur [-1.7] , **sodium [-2.5]** , zinc [-2.8] , salt [-2.9]
	P102	Gordon Scholes is a member of the ____ political party.	Labor	Labour [-1.3] , Conservative [-1.6] , Green [-2.4] , Liberal [-2.9] , **Labor [-2.9]**
	P530	Kenya maintains diplomatic relations with ____.	Uganda	India [-3.0] , **Uganda [-3.2]** , Tanzania [-3.5] , China [-3.6] , Pakistan [-3.6]
	P176	iPod Touch is produced by ____.	Apple	**Apple [-1.6]** , Nokia [-1.7] , Sony [-2.0] , Samsung [-2.6] , Intel [-3.1]
	P30	Bailey Peninsula is located in ____.	Antarctica	**Antarctica [-1.4]** , Bermuda [-2.2] , Newfoundland [-2.5] , Alaska [-2.7] , Canada [-3.1]
	P178	JDK is developed by ____.	Oracle	IBM [-2.0] , Intel [-2.3] , Microsoft [-2.5] , HP [-3.4] , Nokia [-3.5]
	P1412	Carl III used to communicate in ____.	Swedish	German [-1.6] , Latin [-1.9] , French [-2.4] , English [-3.0] , Spanish [-3.0]
	P17	Sunshine Coast, British Columbia is located in ____.	Canada	**Canada [-1.2]** , Alberta [-2.8] , Yukon [-2.9] , Labrador [-3.4] , Victoria [-3.4]
	P39	Pope Clement VII has the position of ____ .	pope	cardinal [-2.4] , Pope [-2.5] , **pope [-2.6]** , President [-3.1] , Chancellor [-3.2]
	P264	Joe Cocker is represented by music label ____.	Capitol	EMI [-2.6] , BMG [-2.6] , Universal [-2.8] , **Capitol [-3.2]** , Columbia [-3.3]
	P276	London Jazz Festival is located in ____.	London	**London [-0.3]** , Greenwich [-3.2] , Chelsea [-4.0] , Camden [-4.6] , Stratford [-4.8]
	P127	Border TV is owned by ____.	ITV	Sky [-3.1] , **ITV [-3.3]** , Global [-3.4] , Frontier [-4.1] , Disney [-4.3]
	P103	The native language of Mammootty is ____.	Malayalam	**Malayalam [-0.2]** , Tamil [-2.1] , Telugu [-4.8] , English [-5.2] , Hindi [-5.6]
	P495	The Sharon Cuneta Show was created in ____.	Philippines	Manila [-3.2] , **Philippines [-3.6]** , February [-3.7] , December [-3.8] , Argentina [-4.0]
ConceptNet	AtLocation	You are likely to find a overflow in a ____.	drain	sewer [-3.1] , canal [-3.2] , toilet [-3.3] , stream [-3.6] , **drain [-3.6]**
	CapableOf	Ravens can ____.	fly	**fly [-1.5]** , fight [-1.8] , kill [-2.2] , die [-3.2] , hunt [-3.4]
	CausesDesire	Joke would make you want to ____.	laugh	cry [-1.7] , die [-1.7] , **laugh [-2.0]** , vomit [-2.6] , scream [-2.6]
	Causes	Sometimes virus causes ____.	infection	disease [-1.2] , cancer [-2.0] , **infection [-2.6]** , plague [-3.3] , fever [-3.4]
	HasA	Birds have ____.	feathers	wings [-1.8] , nests [-3.1] , **feathers [-3.2]** , died [-3.7] , eggs [-3.9]
	HasPrerequisite	Typing requires ____.	speed	patience [-3.5] , precision [-3.6] , registration [-3.8] , accuracy [-4.0] , **speed [-4.1]**
	HasProperty	Time is ____.	finite	short [-1.7] , passing [-1.8] , precious [-2.9] , irrelevant [-3.2] , gone [-4.0]
	MotivatedByGoal	You would celebrate because you are ____.	alive	happy [-2.4] , human [-3.3] , **alive [-3.3]** , young [-3.6] , free [-3.9]
	ReceivesAction	Skills can be ____.	taught	acquired [-2.5] , useful [-2.5] , learned [-2.8] , combined [-3.9] , varied [-3.9]
	UsedFor	A pond is for ____.	fish	swimming [-1.3] , fishing [-1.4] , bathing [-2.0] , **fish [-2.8]** , recreation [-3.1]

Table 3: Examples of generation for BERT-large. The last column reports the top five tokens generated together with the associated log probability (in square brackets).

that BERT-large is able to recall such knowledge better than its competitors and at a level remarkably competitive with non-neural and supervised alternatives. Note that we did *not* compare the ability of the corresponding architectures and objectives to capture knowledge in a given body of text but rather focused on the knowledge present in the weights of existing pretrained models that are being used as starting points for many researchers' work. Understanding which aspects of data our commonly-used models and learning algorithms are capturing is a crucial field of research and this paper complements the many studies focused on the learned linguistic properties of the data.

We found that it is non-trivial to extract a knowledge base from text that performs on par to directly using pretrained BERT-large. This is despite providing our relation extraction baseline with only data that is likely expressing target facts, thus reducing potential for false negatives, as well as using a generous entity-linking oracle. We suspected BERT might have an advantage due to the larger amount of data it has processed, so we added Wikitext-103 as additional data to the relation extraction system and observed no significant change in performance. This suggests that while relation extraction performance might be difficult to improve with more data, language models trained on ever growing corpora might become a viable alternative to traditional knowledge bases extracted from text in the future.

In addition to testing future pretrained language models using the LAMA probe, we are interested in quantifying the variance of recalling factual knowledge with respect to varying natural language templates. Moreover, assessing multi-token answers remains an open challenge for our evaluation setup.

Acknowledgments

We would like to thank the reviewers for their thoughtful comments and efforts towards improving our manuscript. In addition, we would like to acknowledge three frameworks that were used in our experiments: AllenNLP[7], Fairseq[8] and the Hugging Face PyTorch-Transformers[9] library.

References

Dzmitry Bahdanau, Felix Hill, Jan Leike, Edward Hughes, Pushmeet Kohli, and Edward Grefenstette. 2019. Learning to understand goal specifications by

[7] https://github.com/allenai/allennlp
[8] https://github.com/pytorch/fairseq
[9] https://github.com/huggingface/pytorch-transformers

modelling reward. In *International Conference on Learning Representations (ICLR)*.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, pages 238–247.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 2787–2795.

S. R. K. Branavan, David Silver, and Regina Barzilay. 2011. Learning to win by reading manuals in a monte-carlo framework. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 268–277.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. *CoRR*, abs/1704.00051.

Maxime Chevalier-Boisvert, Dzmitry Bahdanau, Salem Lahlou, Lucas Willems, Chitwan Saharia, Thien Huu Nguyen, and Yoshua Bengio. 2018. Babyai: First steps towards grounded language learning with a human in the loop. *CoRR*, abs/1810.08272.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *CoRR*, abs/1901.02860.

Yann N. Dauphin, Angela Fan, Michael Auli, and David Grangier. 2017. Language modeling with gated convolutional networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 933–941.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018a. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018b. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]*. ArXiv: 1810.04805.

Hady Elsahar, Pavlos Vougiouklis, Arslen Remaci, Christophe Gravier, Jonathon Hare, Frederique Laforest, and Elena Simperl. 2018. T-rex: A large scale alignment of natural language with knowledge base triples. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*.

Yoav Goldberg. 2019. Assessing bert's syntactic abilities. *CoRR*, abs/1901.05287.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Rhinehart, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, et al. 2019. Natural questions: a benchmark for question answering research.

Jelena Luketina, Nantas Nardelli, Gregory Farquhar, Jakob Foerster, Jacob Andreas, Edward Grefenstette, Shimon Whiteson, and Tim Rocktäschel. 2019. A Survey of Reinforcement Learning Informed by Natural Language. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, August 10-16 2019, Macao, China.*

Rebecca Marvin and Tal Linzen. 2018. Targeted syntactic evaluation of language models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1192–1202.

R. Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference.

Gábor Melis, Chris Dyer, and Phil Blunsom. 2017. On the state of the art of evaluation in neural language models. *CoRR*, abs/1707.05589.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. Pointer sentinel mixture models. *CoRR*, abs/1609.07843.

Tomas Mikolov and Geoffrey Zweig. 2012. Context dependent recurrent neural network language model. In *2012 IEEE Spoken Language Technology Workshop (SLT), Miami, FL, USA, December 2-5, 2012*, pages 234–239.

Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. 2016. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018a. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 2227–2237.

Matthew E. Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018b. Dissecting contextual word embeddings: Architecture and representation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1499–1509.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

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

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2018. Coqa: A conversational question answering challenge. *CoRR*, abs/1808.07042.

Daniil Sorokin and Iryna Gurevych. 2017. Context-aware representations for knowledge base relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 1784–1789.

Robert Speer and Catherine Havasi. 2012. Representing general relational knowledge in conceptnet 5. In *LREC*, pages 3679–3686.

Mihai Surdeanu and Heng Ji. 2014. Overview of the English Slot Filling Track at the TAC2014 Knowledge Base Population Evaluation. page 15.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4149–4158.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 6000–6010.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the Workshop: Analyzing and Interpreting Neural Networks for NLP, BlackboxNLP@EMNLP 2018, Brussels, Belgium, November 1, 2018*, pages 353–355.

Sean Welleck, Kianté Brantley, Hal Daumé III, and Kyunghyun Cho. 2019. Non-monotonic sequential text generation. *arXiv preprint arXiv:1902.02192*.

Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. 2014. Recurrent neural network regularization. *CoRR*, abs/1409.2329.

Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2018. From recognition to cognition: Visual commonsense reasoning. *CoRR*, abs/1811.10830.

Yukun Zhu, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 19–27.