Unsupervised Improvement of Factual Knowledge in Language Models

Nafis Sadeq*, **Byungkyu Kang**†, **Prarit Lamba**†, **Julian McAuley***
Intuit† and UC San Diego*
{nsadeq,jmcauley}@ucsd.edu
{Jay_Kang,Prarit_Lamba}@intuit.com

Abstract

Masked language modeling (MLM) plays a key role in pretraining large language models. But the MLM objective is often dominated by high-frequency words that are sub-optimal for learning factual knowledge. In this work, we propose an approach for influencing MLM pretraining in a way that can improve language model performance on a variety of knowledge-intensive tasks. We force the language model to prioritize informative words in a fully unsupervised way. Experiments demonstrate that the proposed approach can significantly improve the performance of pretrained language models on tasks such as factual recall, question answering, sentiment analysis, and natural language inference in a closed-book setting.

1 Introduction

Pretrained language models (PLMs) such as BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), BART (Lewis et al., 2020), T5 (Raffel et al., 2020) use a Masked Language Modeling (MLM) objective during pretraining. However, a traditional MLM objective may not be optimal for knowledge-intensive tasks (Peters et al., 2019). It has been shown that language models can benefit from incorporating knowledge within the training objective in the form of entity embeddings (Peters et al., 2019; Zhang et al., 2019), knowledge retriever (Guu et al., 2020), knowledge embedding (Wang et al., 2021; Sun et al., 2020) or augmented pretraining corpora created from Knowledge Graphs (Agarwal et al., 2021). Despite their effectiveness, these approaches rely on existing knowledge bases and entity embeddings to incorporate knowledge within the training objective. These resources are expensive to construct and may not be available for all languages and domains (Huang et al., 2022).

In this work, we propose a pretraining approach that can achieve better performance on knowledge-intensive tasks without using any existing knowledge base. We combine two key strategies to influence MLM objective. Firstly, the tokens with higher informative relevance should be masked more frequently (Sadeq et al., 2022). Secondly, mistakes on informative tokens should be penalized more severely. The informative relevance of the tokens can be computed efficiently with a one-pass computation on the pretraining corpora. Experiments demonstrate that the proposed training strategy can help the language model achieve better performance on the factual knowledge recall benchmark LAMA (Petroni et al., 2019), extractive question answering (QA) benchmark SQuAD (Rajpurkar et al., 2016, 2018), prompt based sentiment analysis and natural language inference (NLI) tasks in AutoPrompt (Shin et al., 2020).

The key contribution of this work is proposing a completely unsupervised stand-alone MLM pretraining objective for language models that can significantly improve performance on knowledge-intensive tasks. Unlike prior works in the area, our method does not require existing knowledge bases to incorporate knowledge during pretraining. We make the code publicly available. [1]

2 Related Work

PLMs as knowledge bases It has been shown that large-scale PLMs such as BERT can be used as a knowledge base (Petroni et al., 2019, 2020). Prior works have focused on factual knowledge with regards to generative PLMs (Liu et al., 2021), multilingual setting (Jiang et al., 2020a), entities and query types (Heinzerling and Inui, 2021), fact checking (Lee et al., 2020).

Designing better prompts Jiang et al. (2020b) propose mining-based and paraphrasing-based methods for automatically generating prompts for improved factual recall performance. A similar

[1]The code is available at https://github.com/intuit/wMLM.git

Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 2960–2969
May 2-6, 2023

approach is explored by Zhong et al. (2021); Haviv et al. (2021); Qin and Eisner (2021). Shin et al. (2020) propose an approach for automatically creating MLM prompts for a diverse range of tasks such as sentiment analysis, natural language inference, relation extraction, etc.

Knowledge integration during pretraining Peters et al. (2019) use entity embeddings from existing knowledge bases and incorporate an entity linking loss jointly with an MLM loss to improve the factual recall performance of BERT. Similarly, Zhang et al. (2019); Wang et al. (2021); Févry et al. (2020); Sun et al. (2020); Liu et al. (2020) use entity representations or knowledge representation from existing knowledge bases to incorporate knowledge into the PLM. Guu et al. (2020) jointly pretrain a knowledge retriever along with a language modeling objective for knowledge integration. Agarwal et al. (2021) synthesize a text corpus from existing knowledge bases and use that during pretraining. Sun et al. (2019) use entity-level and phrase-level knowledge masking during training.

Knowledge modification after pretraining De Cao et al. (2021); Zhu et al. (2020) use constraint optimization for editing existing world knowledge within PLMs with minimal impact on the rest of the factual knowledge. Similarly, Verga et al. (2021) develop a fact injection language model architecture that allows easy integration of existing knowledge bases into PLMs without additional pretraining.

3 Methodology

We use MLM objective for pretraining, similar to prior works (Devlin et al., 2019; Liu et al., 2019; Lewis et al., 2020; Raffel et al., 2020). Given a sequence of tokens Z, a subset of tokens $X \subset Z$ is randomly sampled for replacement ($|X|/|Z| \approx 0.15$ in Devlin et al. (2019)). For the replacement candidates in X, 80% of the time the replacement is done with a special token `[MASK]`, 10% of tokens are replaced with a random token, and the other 10% of candidates are left unchanged (Devlin et al., 2019; Liu et al., 2019; Joshi et al., 2020). The task of the model during pretraining is to predict the original tokens from the modified input sequence. For a set of replaced tokens $X(x_1, x_2, ..., x_N)$ and their corresponding output tokens $Y(y_1, y_2, ..., y_N)$, the loss $\mathcal{L}_{MLM}$ is computed as follows:

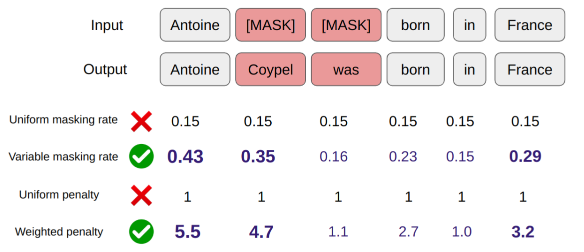

Figure 1: Simplified illustration of variable masking rate and weighted penalty

$$\mathcal{L}_{MLM} = -\sum_{i=1}^{N} \log \frac{e^{x_{i,y_i}}}{\sum_{v \in V} e^{x_{i,v}}} \tag{1}$$

Here, $x_{i,j}$ is the logit produced for output candidate j given input x_i and V is the vocabulary set. In traditional MLM loss computation, a uniform penalty is applied for all tokens within the vocabulary. In our work, we try to influence the MLM objective during pretraining to incorporate more factual knowledge. We differ from traditional MLM pretraining in two ways: **(a)** Instead of masking all tokens with equal probability, we allow some tokens to be masked more frequently if they have higher informative relevance, **(b)** We use weighted cross entropy loss to penalize mistakes on some tokens more severely if they have higher informative relevance. Simple illustrations of these two concepts are shown in Figure 1. We compute the loss as follows:

$$\mathcal{L}_{MLM} = -\sum_{i=1}^{N} w_{y_i} \log \frac{e^{x_{i,y_i}}}{\sum_{v \in V} e^{x_{i,v}}} \tag{2}$$

w_{y_i} is a penalty weight specific to a particular output token y_i. The magnitude of the weight is chosen based on the informative relevance of the tokens. A demonstration of this weighting is shown in Figure 1. Each token in the language model vocabulary has a unique masking rate and penalty weight associated with it. These values can be computed with a one-pass computation before pretraining.

In this context, the informative relevance of tokens represents how important a particular token is with regard to the factual knowledge. Tokens that are more important for factual knowledge (e.g. named entities) are expected to have a higher informative relevance. We use Pointwise Mutual Information (PMI (Fano, 1961)) to compute informative relevance in an unsupervised manner. We hypothesize that words that have high PMI with their

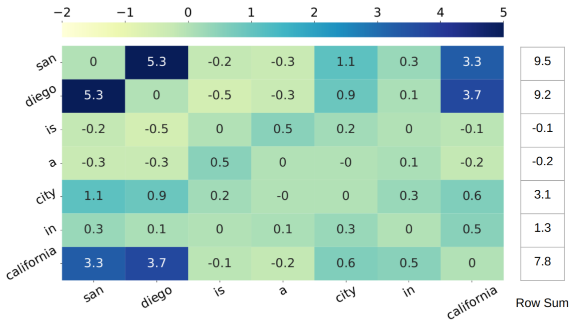

Figure 2: Informative relevance of tokens in a particular document, by computing row-wise summation of the PMI matrix of all token pairs

neighboring words tend to have higher informative relevance. Firstly, we compute word co-occurrence statistics for the pretraining corpus within a skip-gram window. Secondly, PMI between all word pairs within the vocabulary is computed. Thirdly, we consider the pairwise PMI between all words within a particular document in the form of a matrix (as shown in Figure 2), so that the row-wise sum in that matrix reflects the token-specific informative relevance within that document. Then informative relevance for a token is averaged across the corpus. Finally, the computed values are normalized and converted to token-specific masking rates and token-specific penalty weights. Those masking rates are used to create masked inputs and the penalty weights are then incorporated during MLM loss computation, as shown in Equation 2.

4 Experiments

4.1 Pretraining Setup

We use the Wikipedia corpus available in Hugging Face (Lhoest et al., 2021) for pretraining, using a wordpiece tokenizer with a vocabulary size of 100k. The vocabulary size is chosen to ensure the inclusion of most entities. Word co-occurrence statistics are computed using a skip-gram window size of 10. The size of the matrix that holds the PMI between words is $100k \times 100k$. The one-pass computation involving informative relevance of tokens takes around two hours and requires 11 GB of memory. The masking rate for individual tokens varies between 15%-50%, depending on their informative relevance. The average masking rate for all tokens is 19%. The penalty weights for tokens are normalized within the range $[1, 5]$. Training is done with Hugging Face Transformers (Wolf et al., 2020) on an AWS p3.8xlarge machine with 4 Nvidia V100 GPUs. Our model architecture is similar to BERT-base (Devlin et al., 2019) with 12 layers and a hidden dimension of 768. The overall batch size is 128 with a learning rate of 5e-5 and an AdamW optimizer (Loshchilov and Hutter, 2019). Training is done for 10 epochs with a maximum document length of 128. Unlike BERT (Devlin et al., 2019), we do not use the next sentence prediction objective during pretraining. Additionally, the increased masking rate and penalty weight only apply to whole-word tokens. For the subword tokens, we use the minimum masking rate of 15% and penalty weight of 1.

4.2 Evaluation Benchmarks

We use LAMA knowledge probes (Petroni et al., 2019) for evaluating the factual recall performance of the model. LAMA has around 70k samples across 46 factual relations. To evaluate the performance on extractive QA, we use SQuAD v1 and v2 (Rajpurkar et al., 2016, 2018). For zero-shot performance evaluation on closed-book QA, we use the SQuAD portion from LAMA (Petroni et al., 2019). For closed-book sentiment analysis and NLI, we use SST2 and NLI probes from AutoPrompt (Shin et al., 2020). We also report the performance of the models on GLUE (Wang et al., 2018).

4.3 Baselines

We train four models using the same corpus, tokenizer and hyper-parameter setting mentioned in Section 4.1: **(a)** BERT_{uu}: Similar to Devlin et al. (2019), it uses a uniform masking rate and uniform penalty across tokens. This is our baseline. **(b)** BERT_{uw}: uses a uniform masking rate and weighted penalty. **(c)** BERT_{vu} (Sadeq et al., 2022): uses a variable masking rate across tokens and uniform penalty. **(d)** BERT_{vw}: This is our proposed approach that combines both a variable masking rate and weighted penalty across different tokens.

4.4 Results and Discussion

Factual Recall and Zero-shot QA The model using the proposed pretraining approach (BERT_{vw}) significantly outperforms the baseline (BERT_{uu}) on factual recall tasks in LAMA (shown in Table 1). The relative improvement of Mean Reciprocal Rank (MRR) over the baseline is 17.5%, 6%, and 8.1% for ConceptNet, GoogleRE, and TREx respectively. The SQuAD portion of the LAMA benchmark is a set of zero-shot QA samples adapted in a closed-book template. In this task,

Model	LAMA (Petroni et al., 2019)				AutoPrompt (Shin et al., 2020)		
	ConceptNet	GoogleRE	SQuAD	TREx	SST2	NLI (3 way)	NLI (2 way)
BERT_{uu}	0.114	0.281	0.156	0.578	0.651	0.397	0.620
BERT_{uw}	0.120	0.289	0.169	0.592	0.655	0.439	0.676
BERT_{vu}	0.129	0.292	0.175	0.616	0.700	0.457	0.697
BERT_{vw}	**0.134**	**0.298**	**0.187**	**0.625**	**0.704**	**0.481**	**0.711**

Table 1: Factual Recall performance on LAMA, Sentiment Analysis and Natural Language Inference on AutoPrompt. The metrics used for LAMA and AutoPrompt are Mean Reciprocal Rank (MRR) and Accuracy respectively.

Model	SQuAD		GLUE (Wang et al., 2018)								
	v1 (2016)	v2 (2018)	CoLA	SST2	MNLI	QNLI	QQP	STSB	RTE	WNLI	MRPC
BERT_{uu}	69.96	83.22	**31.06**	88.30	79.42	87.72	89.77	85.41	**66.43**	42.25	87.78
BERT_{uw}	71.17	84.17	28.55	89.11	79.82	87.15	89.59	85.70	58.84	49.30	87.93
BERT_{vu}	71.17	85.07	29.11	89.79	80.02	88.21	**90.10**	85.60	61.37	54.93	88.29
BERT_{vw}	**72.61**	**85.28**	28.93	**89.91**	**80.25**	**88.49**	89.82	**85.82**	59.93	**56.34**	**88.32**

Table 2: Performance on SQuAD and GLUE development set. For SQuAD, we report the F1 score. We report the Matthews correlation for CoLA, Pearson correlation for STSB, and accuracy for other GLUE tasks. The fine-tuning parameters for SQuAD and GLUE can be found in Appendix B.

we achieve 19.9% relative improvement over the baseline.

Case studies on factual recall are shown in Table 3. There are two key observations in these case studies. Firstly, the proposed model (BERT_{vw}) is more likely to rank the ground truth label higher during knowledge probes. This helps the model achieve better overall MRR. Secondly, the proposed model is more likely to produce specific words given a particular context when the baseline is only producing generic words. For example, when we use the prompt ‘During Super Bowl 50 the `[MASK]` gaming company debuted their ad for the first time’, the top three candidates from the baseline model are comparatively common words such as ‘computer’, ‘electronic’, and ‘American’. But the proposed model is able to produce more specific words associated with three gaming companies (‘Nintendo, ‘Walt’, and ‘Atari’), including the correct answer ‘Nintendo’. Similar observation can be made with the probe ‘The organization that runs the satellite that measured dust that landed on the Amazon is `[MASK]`’, where the proposed model makes specific predictions with the given context, such as ‘NASA’, ‘Brazil’ and ‘Amazon’. But the baseline can only produce generic words like ‘unknown’, ‘the’, and ‘unclear’.

Closed-book Sentiment Analysis and NLI We use AutoPrompt (Shin et al., 2020) to evaluate the closed-book sentiment analysis and NLI performance of the system. AutoPrompt provides a way to convert certain NLP tasks into a template-based probing format. The advantage of this type of prompting is that it allows us to exploit the factual knowledge within language models without the limitations of fine-tuning (Wallat et al., 2020). The prompt contains the input, a placeholder for the answers, and a span of trigger words (prompt templates shown in Appendix C). The trigger words are tuned using the training dataset and then subsequently used during evaluation. The proposed system achieves 8.1%, 21.1%, and 14.7% relative improvement in accuracy over the baseline in sentiment analysis, 3-way NLI, and 2-way NLI respectively (Table 1).

Fine-tuning vs Prompt-tuning Our proposed model achieves better performance compared to the baseline when fine-tuned on the extractive QA benchmark SQuAD (Rajpurkar et al., 2016, 2018) and text classification benchmark GLUE (Wang et al., 2018). It outperforms the baseline on both SQuAD v1 and v2 tasks and seven out of nine GLUE tasks (shown in Table 2). However, the relative performance improvement with fine-tuning is not as significant as factual recall, zero-shot QA, or prompt-tuning scenarios. The reason behind this may be explained by the findings of Wallat et al. (2020). The main strength of our approach is the ability to store more factual knowledge during pretraining. However, Wallat et al. (2020) have

Input	Ground Truth	$BERT_{uu}$ (Devlin et al., 2019)		$BERT_{vw}$ (proposed)	
		Prediction	Score	Prediction	Score
To emphasize the 50th anniversary of the Super Bowl the [MASK] color was used.	gold	yellow red green	0.17 0.13 0.12	**gold** rainbow orange	0.09 0.06 0.06
During Super Bowl 50 the [MASK] gaming company debuted their ad for the first time.	nintendo	computer electronic american	0.06 0.05 0.03	**nintendo** walt atari	0.05 0.04 0.04
A teacher is most likely teaching at a [MASK].	school	university **school** college	0.61 0.26 0.03	**school** university seminary	0.40 0.34 0.09
Photosynthesis releases [MASK] into the Earth's atmosphere.	oxygen	sunlight photosynthesis light	0.13 0.09 0.09	**oxygen** carbon sunlight	0.21 0.12 0.06
The organization that runs the satellite that measured dust that landed on the Amazon is [MASK] .	nasa	unknown the unclear	0.11 0.03 0.03	**nasa** brazil amazon	0.06 0.05 0.02
Income inequality began to increase in the US in the [MASK].	1970s	1960s 1980s **1970s**	0.21 0.18 0.17	**1970s** 1960s 1980s	0.14 0.13 0.12
He moved to [MASK] at age 16 to complete his high school studies and obtained his Japanese citizenship in 1995.	japan	tokyo **japan** yokohama	0.42 0.21 0.03	**japan** tokyo hawaii	0.19 0.18 0.06
The Crimes Act 1914 is a piece of Federal legislation in [MASK].	australia	canada **australia** england	0.39 0.07 0.03	**australia** tennessee canada	0.12 0.09 0.09
She is also member of the Helsinki City Council and the chairperson of the local party organisation in [MASK].	helsinki	finland **helsinki** espoo	0.52 0.38 0.01	**helsinki** finland espoo	0.76 0.18 0.03
Mark Schwahn (born July 5, 1966) is an American [MASK], director and producer.	screenwriter	actor **screenwriter** writer	0.66 0.14 0.13	**screenwriter** writer actor	0.53 0.21 0.16

Table 3: Case Study from factual recall samples from LAMA (Petroni et al., 2019)

shown that the factual knowledge learned during pretraining may be lost during fine-tuning, limiting the advantage of our proposed system. On the other hand, relational probing, zero-shot QA, and prompt-tuning-based NLP tasks can exploit the additional knowledge of our model more effectively, leading to much better performance.

Ablation Study We investigate how much performance improvement is due to the variable masking rate as opposed to the weighted penalty during MLM pretraining. This can be found by comparing $BERT_{uw}$ with $BERT_{vu}$ (Table 1 and 2). In most cases, we find that a variable masking rate performs slightly better than a weighted penalty.

5 Conclusion

In this work, we propose a pretraining strategy that can be effective in storing factual knowledge within language models. The additional knowledge helps the model outperform previous approaches on a variety of knowledge-intensive NLP tasks, such as factual recall, zero-shot QA, closed-book sentiment analysis, and natural language inference. Our model also achieves better performance when fine-tuned on SQuAD and GLUE tasks. In the future, we aim to extend our work for text-to-text pretrained models such as T5 (Raffel et al., 2020).

Acknowledgements

This work was performed during the first author's internship at Intuit. We thank anonymous reviewers for providing their valuable feedback on this work.

Limitations

One limitation of the proposed system is that it under-performs compared to the baseline in some fine-tuning tasks, such as CoLA (Table 2). The proposed training objective reduces the importance of stopwords in the pretraining objective. This may have a negative impact on performance in tasks where the syntax is important. More investigation is needed to understand and mitigate this issue.

Ethics Statement

A potential concern for the proposed system is that this training strategy may amplify the existing toxic behavior or bias of the language model if the related keywords get prioritized in the training objective. Reducing the toxic or biased behaviors of the proposed model can be an interesting research direction for future work.

References

Oshin Agarwal, Heming Ge, Siamak Shakeri, and Rami Al-Rfou. 2021. Knowledge graph based synthetic corpus generation for knowledge-enhanced language model pre-training. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3554–3565, Online. Association for Computational Linguistics.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Editing factual knowledge in language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6491–6506, Online and Punta Cana, Dominican Republic. 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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Robert M Fano. 1961. Transmission of information: A statistical theory of communications. *American Journal of Physics*, 29(11):793–794.

Thibault Févry, Livio Baldini Soares, Nicholas FitzGerald, Eunsol Choi, and Tom Kwiatkowski. 2020. Entities as experts: Sparse memory access with entity supervision. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4937–4951, Online. Association for Computational Linguistics.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International Conference on Machine Learning*, pages 3929–3938. PMLR.

Adi Haviv, Jonathan Berant, and Amir Globerson. 2021. BERTese: Learning to speak to BERT. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3618–3623, Online. Association for Computational Linguistics.

Benjamin Heinzerling and Kentaro Inui. 2021. Language models as knowledge bases: On entity representations, storage capacity, and paraphrased queries. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1772–1791, Online. Association for Computational Linguistics.

Zijie Huang, Zheng Li, Haoming Jiang, Tianyu Cao, Hanqing Lu, Bing Yin, Karthik Subbian, Yizhou Sun, and Wei Wang. 2022. Multilingual knowledge graph completion with self-supervised adaptive graph alignment. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 474–485, Dublin, Ireland. Association for Computational Linguistics.

Zhengbao Jiang, Antonios Anastasopoulos, Jun Araki, Haibo Ding, and Graham Neubig. 2020a. X-FACTR: Multilingual factual knowledge retrieval from pretrained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5943–5959, Online. Association for Computational Linguistics.

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

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Nayeon Lee, Belinda Z. Li, Sinong Wang, Wen-tau Yih, Hao Ma, and Madian Khabsa. 2020. Language models as fact checkers? In *Proceedings of the Third Workshop on Fact Extraction and VERification (FEVER)*, pages 36–41, Online. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.

Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, et al. 2021. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2020. K-BERT: enabling language representation with knowledge graph. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 2901–2908. AAAI Press.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021. GPT understands, too. *CoRR*, abs/2103.10385.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Matthew E. Peters, Mark Neumann, Robert L. Logan IV, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 43–54. Association for Computational Linguistics.

Fabio Petroni, Patrick S. H. Lewis, Aleksandra Piktus, Tim Rocktäschel, Yuxiang Wu, Alexander H. Miller, and Sebastian Riedel. 2020. How context affects language models' factual predictions. In *Conference on Automated Knowledge Base Construction, AKBC 2020, Virtual, June 22-24, 2020*.

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.

Guanghui Qin and Jason Eisner. 2021. Learning how to ask: Querying lms with mixtures of soft prompts. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 5203–5212. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

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.

Nafis Sadeq, Canwen Xu, and Julian McAuley. 2022. InforMask: Unsupervised informative masking for language model pretraining. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5866–5878, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

Tianxiang Sun, Yunfan Shao, Xipeng Qiu, Qipeng Guo, Yaru Hu, Xuanjing Huang, and Zheng Zhang. 2020. Colake: Contextualized language and knowledge embedding. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 3660–3670. International Committee on Computational Linguistics.

Yu Sun, Shuohuan Wang, Yu-Kun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Danxiang Zhu, Hao Tian, and Hua Wu. 2019. ERNIE: enhanced representation through knowledge integration. *CoRR*, abs/1904.09223.

Pat Verga, Haitian Sun, Livio Baldini Soares, and William Cohen. 2021. Adaptable and interpretable neural MemoryOver symbolic knowledge. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3678–3691, Online. Association for Computational Linguistics.

Jonas Wallat, Jaspreet Singh, and Avishek Anand. 2020. BERTnesia: Investigating the capture and forgetting of knowledge in BERT. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 174–183, Online. Association for Computational Linguistics.

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

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021. Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi 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 M. 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, EMNLP 2020 - Demos, Online, November 16-20, 2020*, pages 38–45. Association for Computational Linguistics.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: enhanced language representation with informative entities. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1441–1451. Association for Computational Linguistics.

Zexuan Zhong, Dan Friedman, and Danqi Chen. 2021. Factual probing is [MASK]: Learning vs. learning to recall. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5017–5033, Online. Association for Computational Linguistics.

Chen Zhu, Ankit Singh Rawat, Manzil Zaheer, Srinadh Bhojanapalli, Daliang Li, Felix X. Yu, and Sanjiv Kumar. 2020. Modifying memories in transformer models. *CoRR*, abs/2012.00363.

A Performance on LAMA by Relation

Domain	Dataset	$BERT_{uu}$	$BERT_{uw}$	$BERT_{vu}$	$BERT_{vw}$
ConceptNet	test	0.114	0.120	0.129	**0.134**
GoogleRE	dateOfBirth	0.099	0.109	0.111	**0.113**
GoogleRE	placeOfBirth	0.456	0.459	0.461	**0.465**
GoogleRE	placeOfDeath	0.288	0.300	0.305	**0.315**
Squad	test	0.156	0.169	0.175	**0.187**
TREx	P1001	0.779	0.770	0.793	**0.798**
TREx	P101	0.442	0.468	0.501	**0.514**
TREx	P103	0.822	0.834	**0.838**	0.836
TREx	P106	0.642	0.653	**0.675**	0.664
TREx	P108	0.491	0.526	0.538	**0.556**
TREx	P127	0.586	0.615	0.620	**0.636**
TREx	P1303	0.380	0.427	0.433	**0.472**
TREx	P131	0.690	0.702	0.741	**0.750**
TREx	P136	0.595	0.629	0.651	**0.675**
TREx	P1376	0.747	0.761	0.783	**0.792**
TREx	P138	0.633	0.640	0.656	**0.680**
TREx	P140	0.569	0.574	**0.608**	0.602
TREx	P1412	0.764	0.773	**0.785**	0.781
TREx	P159	0.535	0.551	0.573	**0.576**
TREx	P17	0.870	0.863	0.884	**0.887**
TREx	P176	0.647	0.673	0.699	**0.720**
TREx	P178	0.569	0.592	0.631	**0.639**
TREx	P19	0.477	0.478	0.509	**0.519**
TREx	P190	0.279	0.276	0.296	**0.297**
TREx	P20	0.511	0.533	0.559	**0.565**
TREx	P264	0.247	0.280	0.291	**0.313**
TREx	P27	0.745	0.756	0.767	**0.773**
TREx	P276	0.625	0.623	0.652	**0.663**
TREx	P279	0.512	0.544	0.562	**0.580**
TREx	P30	0.802	0.813	0.835	**0.842**
TREx	P31	0.616	0.627	0.635	**0.635**
TREx	P36	0.569	0.578	**0.618**	0.615
TREx	P361	0.530	0.538	0.567	**0.574**
TREx	P364	0.703	0.715	0.729	**0.742**
TREx	P37	0.701	0.688	**0.728**	0.715
TREx	P39	0.572	0.607	0.613	**0.630**
TREx	P407	0.638	0.630	0.647	**0.666**
TREx	P413	0.422	0.453	0.483	**0.507**
TREx	P449	0.416	0.444	0.454	**0.495**
TREx	P463	0.646	0.674	0.697	**0.713**
TREx	P47	0.492	0.508	0.564	**0.565**
TREx	P495	0.685	0.662	**0.699**	0.681
TREx	P527	0.423	0.452	0.521	**0.527**
TREx	P530	0.379	0.373	0.400	**0.416**
TREx	P740	0.407	0.414	0.438	**0.438**
TREx	P937	0.528	0.541	0.569	**0.569**

Table 4: Relation by relation performance comparison on LAMA (Petroni et al., 2019)

B Hyper-parameter for fine-tuning on GLUE, SQuAD

Hyper-parameter	GLUE	SQuAD
Batch Size	32	12
Learning Rate	2e-5	3e-5
Epochs	3	2
Weight Decay	0.01	0.01

Table 5: Fine-tuning hyper-parameters for GLUE and SQuAD

C Hyper-parameter for AutoPrompt

Hyper-parameter	SST2	NLI
# Trigger Token	3	4
# Candidate	100	10
Batch Size	24	32
# Iterations	180	100

Table 6: Prompt-tuning hyper-parameters for AutoPrompt (Shin et al., 2020)

Task	Template	Prompt Example	Labels
SST2	{sentence} [T] . . . [T] [P]	director rob marshall went out gunning to make a great one movie director cinema [MASK]	**pos:** partnership, good **neg:** worse, bad
NLI	{prem}[P] [T] . . . [T] {hyp}	There is no man in a black jacket doing tricks on a motorbike [MASK] strange workplace A person in a black jacket is doing tricks on a motorbike	**con:** Nobody, nobody, nor **ent:** found, ways, Agency **neu:** ##ponents, ##lary, ##uated

Table 7: Prompt template for Sentiment Analysis and Natural Language Inference tasks in AutoPrompt (Shin et al., 2020)