SPE: Symmetrical Prompt Enhancement for Fact Probing

Yiyuan Li[*]
UNC-Chapel Hill
yiyuanli@cs.unc.edu

Tong Che[*†]
NVIDIA
tongc@nvidia.com

Yezhen Wang
Mila-Quebec AI Institute
yezhen.wang@mila.quebec

Zhengbao Jiang
Carnegie Mellon University
zhengbaj@cs.cmu.edu

Caiming Xiong
Salesforce Research
cxiong@salesforce.com

Snigdha Chaturvedi
UNC-Chapel Hill
snigdha@cs.unc.edu

Abstract

Pretrained language models (PLMs) have been shown to accumulate factual knowledge during pretraining (Petroni et al., 2019). Recent works probe PLMs for the extent of this knowledge through prompts either in discrete or continuous forms. However, these methods do not consider symmetry of the task: object prediction and subject prediction. In this work, we propose Symmetrical Prompt Enhancement (SPE), a continuous prompt-based method for factual probing in PLMs that leverages the symmetry of the task by constructing symmetrical prompts for subject and object prediction. Our results on a popular factual probing dataset, LAMA, show significant improvement of SPE over previous probing methods.

1 Introduction

Prompt-based learning proposes to formulate different NLP tasks into language modeling problems (Schick and Schütze, 2021). It is a novel paradigm that effectively uses Pretrained Language Models (PLMs) (Liu et al., 2022), and achieves comparable or better performance than fine-tuning (Lester et al., 2021). Prompt-based learning has also been used for the task of factual knowledge probing in PLMs. In this task, the goal is to predict the (masked) object of factual tuples of type (subject, relation, object) using PLMs. Prompting methods assume that PLMs gather and store factual knowledge during their pre-training, and cloze-style prompts can be used to probe PLMs to gauge how much knowledge they contain (Petroni et al., 2019). The prompts are either handcrafted (Petroni et al., 2019; Bouraoui et al., 2020) or automatically generated (Shin et al., 2020; Haviv et al., 2021). For example, to probe PLMs about their knowledge of geographic location of *Luxembourg*, a `prompt` can be formed by filling *Luxembourg* in the first blank of the following `template`: "____ is located in ____.". An effective prompt will probe the PLM to output *Europe* as the most likely prediction for the second blank. Such methods are promising but brittle. Minor changes in the template can lead to significant difference in the performance (Jiang et al., 2020). Recent works have shown that continuous prompts obtained via gradient-based learning, are more effective and robust than discrete prompts since there are less restrictions on the search space (Liu et al., 2021; Qin and Eisner, 2021; Zhong et al., 2021; Liu et al., 2022; Newman et al., 2022).

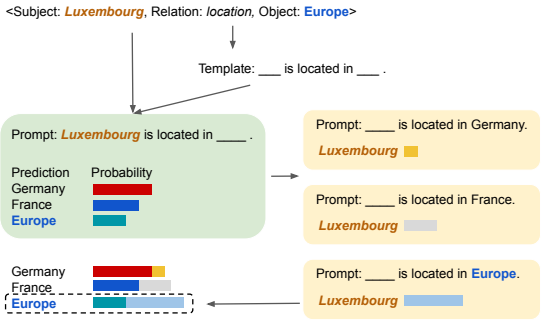

Figure 1: Example of factual probing: Given a subject and relation, predict the object. SPE uses a fixed template to generate a prompt for predicting object given subject (green box) as well as several symmetrical prompts for predicting the subject given object candidates (yellow boxes). The final prediction is obtained using the likelihoods of the object candidates and of the given subject as obtained using the symmetrical prompts. Bars represent probabilities from BERT. SPE is a continuous prompt method but we use natural language prompts and template here for illustration.

Existing methods for learning prompts do not leverage the symmetry inherent in the task's definition. For example, while *Luxembourg* is located in *Europe*, *Europe* contains *Luxembourg*. Similar ideas have been used for learning prompts for relation classification (Han et al., 2021) and other NLP tasks (Crawford et al., 1996; Kiddon and Domingos, 2015; He et al., 2017; Tanchip et al., 2020).

[*] Equal contribution.
[†] Work was done at MILA.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 11689 - 11698
December 7-11, 2022

In this work, we propose *Symmetrical Prompt Enhancement* (SPE)– a continuous prompting method that *learns* prompt that incorporates the above mentioned symmetry. Specifically, in addition to generating a prompt to predict the object given the subject, SPE also generates an additional symmetrical prompt to predict the subject given the object. Using the first prompt (see green box in Fig. 1), SPE obtains a few high-probability candidate objects like *Germany*, *France*, and *Europe*. Thereafter, for each object candidate, it generates a symmetrical prompt (see yellow boxes), and obtains the likelihood of the subject, *Luxembourg*. At the heart of SPE is a prompt generation model that is trained by maximizing the joint likelihood of both the candidates as well as the subject (given the candidates). Our experiments on the factual probing dataset LAMA (Petroni et al., 2019) show that SPE achieves significant improvement over previous approaches and our analysis points to sources of this performance gain. These experiments demonstrate that like SPE, probing methods should learn prompts that leverage the symmetry of the task because that can help PLMs in producing better answers when they are being probed for stored factual knowledge.

2 Symmetrical Prompt Enhancement

The goal of factual probing via prompt generation is to output object $\mathcal{O}$ for given subject $\mathcal{I}$ and relation $\mathcal{R}$ by constructing a prompt $\mathcal{P}$. Most methods operate by assuming a template $\mathcal{T}$, and generating the prompt $\mathcal{P}$ from $\mathcal{T}$, $\mathcal{I}$ and $\mathcal{R}$. Fig. 1 shows an example of Subject (*Luxembourg*), Relation (*location*), Object (*Europe*), Template (____ *is located in* ____.), and the corresponding Prompt (*Luxembourg is located in* ____.). The figure shows a natural language template and prompts for readability. However, for continuous prompt methods like ours, the template is a sequences of vectors like $[V]_1 \dots [V]_n$ ____ $[V]_{n+1} \dots [V]_{n+m}$ ____ $[V]_{n+m+1} \dots [V]_{n+m+k}$, $\forall [V]_i \in \mathbb{R}^d$. We refer to the two blanks as $\mathrm{B}_{\mathcal{O}}$ and $\mathrm{B}_{\mathcal{I}}$. The prompt, $\mathcal{P}_{orig}$, is typically generated by learning these vectors and filling the (representation of) $\mathcal{I}$ in $\mathrm{B}_{\mathcal{I}}$. The prompts are relation-specific ($\mathcal{P}^{\mathcal{R}}_{orig}$) but here we refer to them as $\mathcal{P}_{orig}$ for simplicity. The model's prediction, $\hat{\mathcal{O}}$, is the most likely object candidate for the $\mathrm{B}_{\mathcal{O}}$ as determined by the PLM using $\mathcal{P}_{orig}$.

Our proposed approach, *Symmetrical Prompt Enhancement* (SPE), leverages the inherent symmetry of the task. Specifically, in addition to learning the original prompt $\mathcal{P}_{orig}$ for predicting the object given the subject, SPE also generates several symmetrical prompts, $\mathcal{P}_{sym}$, for predicting the subject given the object. Like $\mathcal{P}_{orig}$, $\mathcal{P}_{sym}$ is also generated from $\mathcal{T}$ except that this time $\mathrm{B}_{\mathcal{O}}$ is filled by the (representation of) $\mathcal{O}$. The prompt is used for probing the PLM which outputs prediction for $\mathrm{B}_{\mathcal{I}}$.

$$p(v|\mathcal{P}_{\mathrm{orig}}) = \mathrm{P}_{\mathrm{PLM}}(\mathrm{B}_{\mathcal{O}} = v|\mathcal{P}_{\mathrm{orig}}) \quad (1)$$

$$p(v'|\mathcal{P}_{\mathrm{sym}}) = \mathrm{P}_{\mathrm{PLM}}(\mathrm{B}_{\mathcal{I}} = v'|\mathcal{P}_{\mathrm{sym}}) \quad (2)$$

Here $p(v|\mathcal{P})$ is the probability distribution of word or phrases v in PLM given prompt $\mathcal{P}$ as input.The model is trained by optimizing a linear combination of the cross-entropy objectives of predicting the object $\mathcal{O}$ and the subject $\mathcal{I}$:

$$\max_{\theta} \log p(v = \mathcal{O}|\mathcal{P}_{orig}) + \lambda \log p(v' = \mathcal{I}|\mathcal{P}_{sym}), \quad (3)$$

where λ is a hyperparameter. θ, the parameters of the prompt generation model, are learned.

For inference, SPE selects top K predictions $\mathcal{C}^K$:

$$\mathcal{C}^K = \mathrm{Top}K_{v \in \mathcal{V}}\, p(v|\mathcal{P}_{\mathrm{orig}}) \quad (4)$$

and uses each prediction $c^k \in \mathcal{C}^K$ as a candidate to generate the symmetrical prompt $\mathcal{P}^k_{sym}$. Finally, the model's prediction $\hat{\mathcal{O}}$ is:

$$\hat{\mathcal{O}} = \arg\max_{c^k \in \mathcal{C}^K} \log p(v = c^k|\mathcal{P}_{\mathrm{orig}}) + \lambda \log p(v' = \mathcal{I}|\mathcal{P}^k_{\mathrm{sym}}). \quad (5)$$

In practice, $\mathcal{L}$ and $\mathrm{P}_{\mathrm{PLM}}$ are normalized by input length to account for inputs with multiple tokens.

3 Implementation Details

We conduct experiments on the fact retrieval part of LAMA dataset (Petroni et al., 2019), which consists of fact triples with single-token objects from 41 relations in Wikidata (Vrandečić and Krötzsch, 2014). We use the training set extended by Shin et al. (2020). We choose masked language models BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) as PLMs, which are fixed during training to serve as static knowledge bases. For implementation, we use PLMs in Huggingface library of Transformers (Wolf et al., 2020). We follow Liu et al. (2021) for designing templates and the prompt generation component of our model. In particular, we use BiLSTM (Graves et al., 2013) with multilayer perceptron (MLP) for prompt generation

Model	BERT-base			BERT-large			RoBERTa-base		
	P@1	P@10	MRR	P@1	P@10	MRR	P@1	P@10	MRR
Manual (Petroni et al., 2019)	31.1	59.5	40.3	28.9	57.7	38.7	22.0	36.0	25.0
LPAQA (Jiang et al., 2020)	34.1	62.0	43.6	39.4	67.4	49.1	21.7	36.0	27.7
AutoPrompt (Shin et al., 2020)	43.3	73.9	53.9	41.3	69.3	50.6	40.0	68.3	49.9
OptiPrompt (manual) (Zhong et al., 2021)	48.6	79.0	58.9	50.6	79.2	60.7	40.3	65.7	48.9
SoftPrompt (mined) (Qin and Eisner, 2021)	48.8	79.6	59.4	51.0	81.4	59.6	40.6	75.5	53.0
P-tuning (Liu et al., 2021)	48.2	78.1	58.6	49.9	80.6	60.6	43.5	73.9	53.8
SPE	**50.3**	**80.5**	**60.9**	**53.1**	**82.4**	**63.4**	**47.0**	**75.8**	**56.2**

Table 1: SPE outperforms state-of-the-art discrete and continuous prompt approaches on the LAMA dataset.

Model	P@1	P@10	MRR
P-tuning	48.2	78.1	58.6
SPE K=1	48.7	79.9	59.5
K=5	49.9	79.9	60.5
K=10	49.9	79.9	60.7
K=15	**50.3**	**80.5**	**60.9**

Table 2: Effect of varying size of candidate pool on SPE's performance. SPE outperforms P-tuning even without reranking (K=1). A larger candidate pool helps the model even further.

and use the following generic and relation-agnostic format for template, $\mathcal{T}$: $[V]_1$ $[V]_2$ $[V]_3$ ____ $[V]_4$ $[V]_5$ $[V]_6$ ____ $[V]_7$ $[V]_8$ $[V]_9$ $\forall [V]_i \in \mathbb{R}^d$. The model and the template are randomly initialized.

For $\mathcal{I}$ with multiple tokens, we mask them one token at a time to generate $\mathcal{P}_{\text{sym}}$, and use the average of pseudo likelihoods from all $\mathcal{P}_{\text{sym}}$s to represent $\log p(v' = \mathcal{I}|\mathcal{P}_{\text{sym}})$. In practice, we find that masking one token at a time is better than masking the entire phrase at once, and averaging the pseudo-likelihood has better performance. The training batch size is 8. We set K to be 15 during inference, and λ to be 0.8 based on our experiments on the development set. The results are evaluated by accuracy at top 1 (P@1) and top 10 (P@10) predictions, and Mean Reciprocal Rank (MRR) as in Qin and Eisner (2021). Appendix A includes more setup details and discussion on choice of λ.

4 Results

We compare our results with both discrete and continuous prompt methods. Discrete prompt methods include prompts from manually designed templates (Petroni et al., 2019); LPAQA (Jiang et al., 2020), which uses text mining based prompts; and AutoPrompt (Shin et al., 2020), which uses discrete lexicalized trigger tokens for prompt generation. Continuous prompt methods include P-tuning (Liu et al., 2021), which uses a neural network to generate prompts; OptiPrompt (Zhong et al., 2021), which uses manually initialized prompts; and SoftPrompt (Qin and Eisner, 2021), which ensembles multiple prompts initialized with mined templates.

Quantitative Results: Table 1 shows the performance of SPE and all baselines. The results show that SPE outperforms all previous methods. Note that, unlike OptiPrompt and SoftPrompt, SPE does not make use of manually designed templates for initialization. We also find that SPE outperforms the baselines when the PLM parameters are updated jointly with the prompt tokens on the training data. See Table 5 in the Appendix B.2 for detailed results. For the rest experiments, we consider P-tuning as primary baseline since it is the best performing model that is directly comparable to SPE.

Effect of candidates pool size: Table 2 shows how SPE performs with different candidate pool sizes. Comparing the first two rows we can see that SPE outperforms our primary baseline, P-tuning, even without reranking (K=1). Increasing the size of the candidate pool leads to further improvements. However, expanding the candidate pool has a trade-off between performance and memory usage. Meanwhile, applying reranking on the discrete prompt methods mentioned does not introduce performance gain, mainly because their prompt templates are selected or mined in favor of object prediction only. We leave the investigation of constructing discrete prompts that benefits from the symmetry as future work.

Performance on Easy and Hard examples: The

Relation	Subject	Top 5 Predictions (Prob. High ⟶ Low): Top - PT, Bottom - SPE					Rank
P108 (employer)	Spike Milligan	Microsoft	IBM	Google	BBC	ESPN	4
		BBC	Microsoft	CBS	ESPN	Google	1
P364 (original language)	Baaz	Turkish	English	French	Arabic	Persian	41
		Hindi	Urdu	Punjabi	Bengali	Persian	1
P101 (field of work)	Richard Wagner	music	history	psychology	opera	linguistics	4
		opera	music	philosophy	aesthetics	art	1
P27 (country of citizenship)	Rubens Barrichello	Belgium	France	Italy	Spain	Germany	15
		Brazil	Spain	Argentina	Portugal	Uruguay	1
P30 (continent)	Marshall Islands	Antarctica	Asia	Africa	Oceania	Europe	4
		Asia	Oceania	Africa	Antarctica	Europe	2
P279 (subclass of)	river	river	stream	tributary	canal	creek	1
		tributary	stream	river	creek	tributaries	3

Table 3: Sample outputs of P-tuning (PT) and SPE. The ranks of correct answers (underlined) are in the last column.

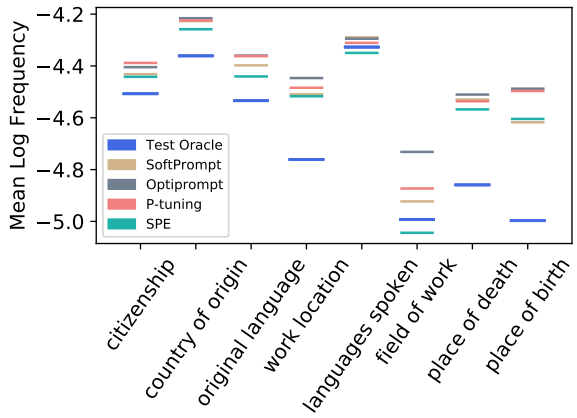

Figure 2: Frequencies of predictions of different methods. SPE can output answers that have low frequencies.

LAMA test set has also been split into LAMA-Easy and LAMA-Hard where objects in the LAMA-Easy split can be "guessed" by naive methods (Zhong et al., 2021). We observe that SPE outperforms the baselines in P@1 for both splits and its gain over P-tuning for LAMA-Hard (4.2%) is larger than LAMA-Easy (1.5%) (see Table 4 of Appendix B.1). This indicates that the improvement of SPE does not simply come from shallow pattern matching and it performs well on hard examples.

Qualitative Results and Analysis: We include some qualitative examples of top 5 predictions in Table 3 from P-tuning (top half of each row) and SPE (bottom half of each row). The correct answers are underlined, and their ranks in the predicted lists are in the last column. We observe that SPE's top predictions are in the correct domain. For example, SPE outputs *BBC* for the *employer* of British-Irish actor *Spike Milligan* (as opposed to *Microsoft*, *IBM*, and *Google*) as outputted by P-tuning), and *Hindi* along with other Indian languages, when asked about the *original language* of an Indian movie *Baaz*, rather than *Turkish*, a non-Indian Language. Moreover, SPE correctly identifies the *country of citizenship* for *Rubens Barrichello* as *Brazil*. Identifying objects for relations like *country of citizenship* for individuals are challenging because documents with the individual's names in the pretraining corpus of PLMs might contain mentions of multiple places he/she has worked or lived or received education in. Therefore, these co-occurrences might confuse PLMs. In Appendix C, we identify such *confusing* relations and conduct a close analysis on them.

We also find that SPE's predictions (e.g. *opera* for the field of work of *Richard Wagner*) are more precise that P-tuning's (*music*). In general, PLMs predictions for a relation can get affected by related high-frequency but incorrect object candidates. Previous prompt methods are found to suffer from bias of the prompt and object distribution in the dataset (Cao et al., 2021). To investigate this, we identify a set of relations that are prone to such spurious frequency-related associations (see Appendix C for the list of relations) and find that SPE especially performs well on such relations (see Figure 3 and Appendix C.1). We also plot the mean ($\log_{10}$) token frequencies of top predictions of different methods as well as the oracle for these relations in Figure 2 (using word frequencies from Speer et al. (2018)). We observe that SPE's predictions (green bars) have lower frequencies than most baselines including P-tuning (red bars). Meanwhile, the frequencies of SPE's predictions

are in general more similar to that of the oracle (blue bars) than most baselines. This indicates that even though the correct (and more precise) answers have lower frequencies, SPE can output them as answers while the baselines output the more frequent alternatives as answers (see Appendix C.2 for examples). We further extend this analysis to the top M predictions and observe similar behavior (see Appendix C.3). Lastly, the outputs of SPE are less affected by the most frequently occurring objects in the dataset (see Appendix C.4).

5 Limitations

We note that SPE may not help if the correct objects are broad concepts (e.g. "mathematics" vs "algebra", "river" vs "tributary", "FIFA" vs "UEFA"). Typical relations with such objects include P279 *subclass of*, P361 *part of* and P463 *member of*. The top 5 predictions by SPE (and also P-tuning) for the *subclass of* relation are shown in Table 3. The correct answer, *river*, is ranked 3rd by SPE and an incorrect answer, *tributary*, is the top prediction. P-tuning outputs the correct answer.

Also, in general, SPE can get affected by error propagation because of its two-step inference process that first predicts object candidates and then ranks them.

Though the proposed symmetrical prompt method improves knowledge probing, the utility of the technique in other NLP tasks is not yet investigated. Besides, the experiments are only conducted for masked language models but there has been recent progress in other types of language models which are not explored in the paper. Lastly, the proposed method requires additional computational cost compared the baselines.

6 Conclusion

This work introduces Symmetrical Prompt Enhancement (SPE) – a continuous prompt-learning method for factual probing of PLMs by learning prompts that utilize the inherent symmetry of the task. Our experiments show that SPE outperforms existing SOTA methods thereby helping us know more about how much knowledge is stored in a PLM. Future work could explore this idea of using task symmetry for other NLP tasks. [1]

[1]Code is available here.

7 Ethical Consideration

In this work, we propose SPE, which incorporates the symmetrical nature of factual knowledge in prompt methods. Our result shows the effectiveness of SPE over several previous prompt baselines. Even though we work on the factual knowledge dataset, we notice that current PLMs does not have the awareness to distinguish between publicly-available factual knowledge and private information (which is not considered as knowledge) either during the pre-training or inference, while the memorizing information of PLMs in latter lead to potential risk of privacy leakage (Carlini et al., 2021). All the experiments are conducted on the publicly available dataset, which is mainly based on Wikidata.

References

Zied Bouraoui, Jose Camacho-Collados, and Steven Schockaert. 2020. Inducing relational knowledge from bert. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7456–7463.

Boxi Cao, Hongyu Lin, Xianpei Han, Le Sun, Lingyong Yan, Meng Liao, Tong Xue, and Jin Xu. 2021. Knowledgeable or educated guess? revisiting language models as knowledge bases. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1860–1874, Online. Association for Computational Linguistics.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, Alina Oprea, and Colin Raffel. 2021. Extracting training data from large language models. In *USENIX Security Symposium*.

Tyler A. Chang and Benjamin K. Bergen. 2022. Word acquisition in neural language models. *Transactions of the Association for Computational Linguistics*, 10:1–16.

James M. Crawford, Matthew L. Ginsberg, Eugene M. Luks, and Amitabha Roy. 1996. Symmetry-breaking predicates for search problems. In *Proceedings of the Fifth International Conference on Principles of Knowledge Representation and Reasoning*, KR'96, page 148–159, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

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.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 315–323, Fort Lauderdale, FL, USA. PMLR.

Alex Graves, Abdel rahman Mohamed, and Geoffrey E. Hinton. 2013. Speech recognition with deep recurrent neural networks. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649.

Xu Han, Weilin Zhao, Ning Ding, Zhiyuan Liu, and Maosong Sun. 2021. Ptr: Prompt tuning with rules for text classification. *ArXiv*, abs/2105.11259.

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.

He He, Anusha Balakrishnan, Mihail Eric, and Percy Liang. 2017. Learning symmetric collaborative dialogue agents with dynamic knowledge graph embeddings. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1766–1776, Vancouver, Canada. Association for Computational Linguistics.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. How Can We Know When Language Models Know? On the Calibration of Language Models for Question Answering. *Transactions of the Association for Computational Linguistics*, 9:962–977.

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

Chloé Kiddon and Pedro M. Domingos. 2015. Symmetry-based semantic parsing.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2022. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.* Just Accepted.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021. Gpt understands, too. *ArXiv*, 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. *ArXiv*, abs/1907.11692.

Benjamin Newman, Prafulla Kumar Choubey, and Nazneen Rajani. 2022. P-adapters: Robustly extracting factual information from language models with diverse prompts. In *International Conference on Learning Representations*.

Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. 2018. Analyzing uncertainty in neural machine translation. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 3956–3965. PMLR.

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*, pages 5203–5212, Online. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. 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.

Robyn Speer, Joshua Chin, Andrew Lin, Sara Jewett, and Lance Nathan. 2018. Luminosoinsight/wordfreq: v2.2.

Chelsea Tanchip, Lei Yu, Aotao Xu, and Yang Xu. 2020. Inferring symmetry in natural language. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2877–2886, Online. Association for Computational Linguistics.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85.

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.

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.

A Additional Implementation Details

Prompt Generation Model: The prompt generation model is based on work by Liu et al. (2021). It consists of a two-layer BiLSTM and a two-layer MLP on top of it. The MLP uses ReLU (Glorot et al., 2011) as the activation function. The hidden size of LSTM and dimension of d are 768 for BERT-base-cased and RoBERTa-base, and 1024 for BERT-large-cased. The max training epoch is 100, and training stops when development performance does not increase for 20 epochs. The optimizer is Adam with learning rate being 1e-5. Other setting also follows Liu et al. (2021). The number of parameters is determined by the PLMs: BERT-base-cased (110M), BERT-large-cased (340M) and RoBERTa-base (125M); and the prompt generation model (14M). The experiments require 20 hours to finish on a single Tesla V100 GPU.

Also, during our experiments, we experiment with having separate prompt generation models for generating $\mathcal{P}_{orig}$ and $\mathcal{P}_{sym}$. However, we find that training one prompt generation model for both $\mathcal{P}_{orig}$ and $\mathcal{P}_{sym}$ led to better results.

Choice of λ: In our preliminary experiments on the development set, we find $\lambda = 0.8$ to be the best choice among $[0, 1]$. However, we observe that the performance is not very sensitive to λ and $\lambda > 0.4$ generally gives a reasonable performance.

B Additional Results

B.1 Easy and Hard LAMA Examples

Zhong et al. (2021) points out that during factual probing, a PLM's predictions can be based on shallow patterns in the training data instead of the knowledge stored in the PLM. To study this phenomena, they propose an *easy* (LAMA-Easy) and a *hard* (LAMA-Hard) split of the LAMA dataset where objects in the LAMA-Easy subset can be "guessed" by naive or non-pretrained models. We compare SPE with the baselines on these two subsets and report results in Table 4. We observe that, in general, all methods achieves better performance in LAMA-Easy that the complete testset but SPE has the highest P@1. It is outperformed on P@10 and MRR only by Softprompt and Optiprompt but they use manually designed templates. More importantly, SPE shows higher improvement in LAMA-Hard compared to baselines especially with respect to P-tuning (4.2% in P@1). This shows that the improvement of SPE does not simply come from

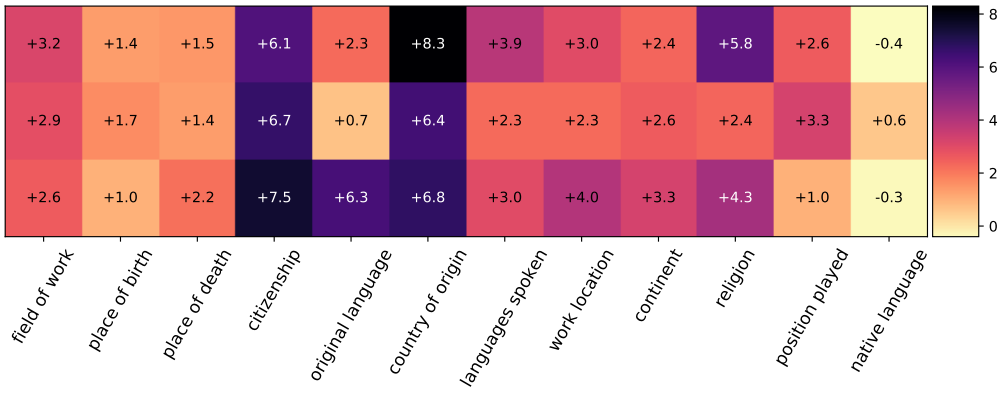

Figure 3: P@1 improvement of SPE under different relations in BERT-large-cased (scale of 100, darker color represents higher value): P101 *field of work* (N-M), P19 *place of birth* (N-1), P20 *place of death*, P27 *citizenship* (N-M), P364 *original language of film or TV show* (N-1), P495 *country of origin* (N-1), P1412 *language spoken* (N-M), P937 *work location* (N-M), P30 *continent* (N-1), P140 *religion* (N-1), P413 *position played* (N-1), and P103 *native language* (N-1). The first, second and third row represents the P@1 improvement SPE has over Optiprompt, SoftPrompt and P-tuning respectively. SPE outperforms all three probing methods for most relations.

shallow pattern matching and it is better at handling more challenging knowledge probing cases.

B.2 Finetuning PLMs

In the experiments reported in the paper, the PLMs are fixed during training and only the prompt generation model is being trained. We now experiment with also finetuning the PLMs. We use BERT-large-cased for this experiment. The results are reported in Table 5 where we compare SPE with the comparable gradient-based baselines. We can see that SPE outperforms those baselines in this setting also. The drop in P@10 of AutoPrompt compared to its P@10 when PLM is fixed (see Table 1) may be related to its discrete token substitution (non-gradient-descent) design, which is harder to optimize.

C Analysis on Relations with Spurious Associations

Recent works have shown that frequency bias exists in maximal likelihood estimation training of language models (Ott et al., 2018; Jiang et al., 2021) and how a PLM's learning of a word is related to its frequency (Chang and Bergen, 2022). Cao et al. (2021) observed that prompts for fact probing overfit the object distribution more than the relation. As a result, in factual probing, PLM's output might get affected by the frequencies of output candidates. This is especially true for relations that are prone to spurious associations of the subject with candidate objects or over-representation of candidate objects. Below, we identify some such relations and then analyze performance of SPE with respect to the baselines on these relations.

R1 Relations with scope associations (P101 *field of work*). When probing factual knowledge from PLMs, the object of a subject-relation pair forms the correct answer. While there can be multiple reasonable answers, some are more precise and so more desirable than others. For instance, for describing the *field* that *Richard Wagner* worked in (see Table 3), both *opera* and *music* seem to be reasonable answers but *opera* is the more precise one. In such relations, different object-candidates may entail similar meanings but be of different scope.

R2 Relations with entity-type associations (P19 *place of birth*, P20 *place of death*, P27 *country of citizenship*, P364 *original language of film or TV show*, P495 *country of origin*, P1412 *language spoken*, P937 *work location*). Some relations are about objects with specific constraints. For example, *place of birth* and *place of death* are the first

Dataset	LAMA-Easy			LAMA-Hard		
Model	P@1	P@10	MRR	P@1	P@10	MRR
Manual	40.2	70.7	47.9	27.1	54.2	35.1
LPAQA	46.0	71.7	52.6	27.3	55.7	35.3
AutoPrompt	58.2	85.7	66.2	28.6	60.9	39.5
Optiprompt	73.2	94.8	81.3	37.6	73.6	50.4
SoftPrompt	77.0	**96.0**	**83.9**	38.4	76.8	51.7
P-tuning	75.9	94.1	82.9	35.2	75.3	49.2
SPE	**77.4**	94.4	83.6	**39.4**	**77.5**	**52.9**

Table 4: SPE outperforms the baselines on both hard and easy examples with greater improvement on the hard ones.

Model	P@1	P@10	MRR
AutoPrompt	41.3	61.6	50.6
Optiprompt	53.3	74.9	63.3
SoftPrompt	51.6	81.9	62.1
P-tuning	51.4	82.1	61.8
SPE	**53.7**	**83.0**	**63.9**

Table 5: Performance comparison of gradient-based prompt methods when the PLM is finetuned. SPE outperforms the baselines.

and last place in a person's life. Those objects, as well as other objects of same entity types that do not match such constraints (e.g. general location names), can co-occur with the subject in the training corpora and get memorized by the pretrained models. Because of these co-occurrences, PLMs may output incorrect objects that are of the correct entity type but may not satisfy the desired constraints. For example, when probed for *place of birth* of an individual, they may output places where the individual received education or worked instead of where they were born. In the example in Table 3, when probing for *citizenship* of famous Brazilian Formula One player *Rubens Barrichello*, P-tuning outputs a handful of countries listed on his Wikipedia page where he participated competitions, which are unrelated to the country of his citizenship, *Brazil*.

R3 Relations with label distribution associations (P30 *continent*, P140 *religion*, P413 *position played*, P103 *native language*) Zhong et al. (2021) showed the label distribution effects prompt-based methods. In particular, for relations with a closed set of candidate objects, the task of factual probing reduces to a classification problem with fixed number of labels. When the correct label (object) appears with very low frequency, PLM's output can get affected by label distribution in the training set and it can output other labels that appear more frequently. For example, in P30 *continent*, 95.6% continent-type objects in the training set are *Antartica* (majority class) and only 0.4% are *Oceania* (minority class). P-tuning is probably affected by this imbalance and outputs the majority label, *Antartica*, as the continent that contains *Marshall Islands* while *Oceania*, the correct answer, appears at rank 4 (see Table 3).

C.1 Comparison of SPE with Baselines on Relations with Spurious Associations

We observe that for R1, R2 and R3 category relations, SPE especially outperformed the baselines in most cases (see Figure 3). The first, second and third rows of the figure represent the corresponding P@1 improvement (scale of 100) of SPE over Optiprompt, SoftPrompt and P-tuning respectively and a darker color means larger improvement.

C.2 Investigating Token Frequencies of Top-1 Predictions.

To further investigate these improvements, we explored the correlation between predictions of different prompt approaches and their token frequencies. We analyzed relations affected by co-occurences of subjects with spurious object candidates, i.e. relations of type R1 and R2. We acquired word frequencies from Speer et al. (2018) who collected word frequencies from 8 domains including Wikipedia, books, and news. As discussed in Section 4 of the paper, we plotted the mean token frequencies of top predictions obtained using different prompting approaches and showed that SPE's predictions have lower frequencies than most baselines. For exam-

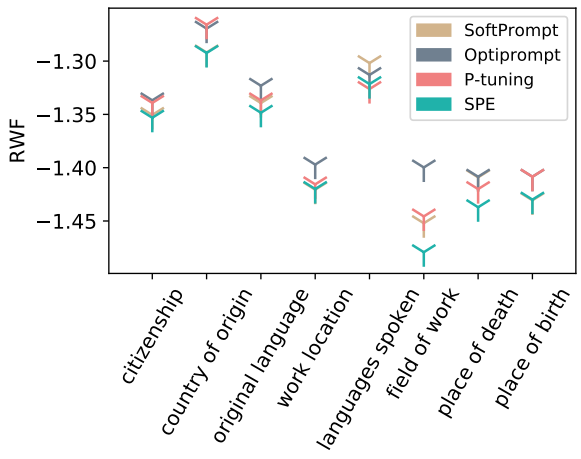

Figure 4: Comparison of different methods using RWF of top 10 predictions for relations with spurious associations. Some markers are not visible because they overlap with the green ones. A lower RWF is better and indicates less association between token frequency and predictions. We see that SPE (green) has lower RWF than P-tuning (pink) in most relations. This indicates that SPE can help PLM in outputing less frequent but correct answers.

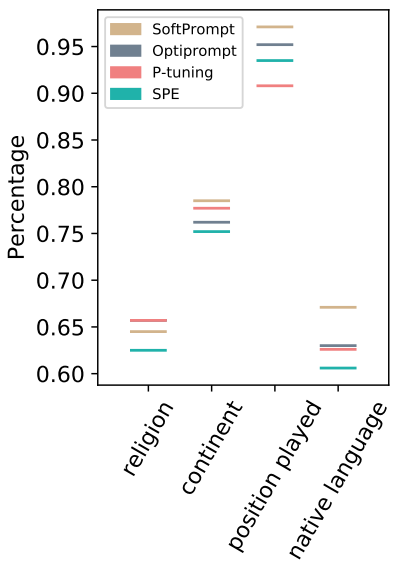

Figure 5: Comparison of different methods in percentage of majority training label in test predictions for R3 relations. A lower value means the method is less affected by the label distribution in the training set. Invisible bars are overlapped with the red ones. SPE (green) has lower values than the baselines in most relations.

ple, in the *field of work* of *Richard Wagner* on Table 3, the log word frequency of *opera* is -4.73 but the log frequencies of *music*, *history*, *philosophy* and *psychology* are -3.48, -3.61, -4.51 and -4.69 respectively, which have higher word frequencies than *opera* (especially, the frequency of P-tuning's output *music* is 17 times higher). Similarly, in the case of *original language* of *Bazz*, the log frequencies of *Hindi*, *Urdu*, *Punjabi* are -5.18, -5.74, -5.84, while for non-Indian languages like *Turkish*, *English*, and *French* they are -4.71, -3.81 and -3.91, which means these frequencies are at least 10 times higher than the Indian languages. Yet, SPE outputs the correct, even though less frequent answers.

C.3 Investigating Token Frequencies of Top-M Predictions.

We now extend the above-mentioned analysis from top predictions to top M predictions and analyze if SPE can help the PLMs output less frequent tokens as answers. In particular, for different prompting approaches, we consider their top M predictions and compute the Rank Weighted Frequency (RWF) using the following formula, where $\mathcal{C}_n^M$ is the n-th candidates among the top M predictions.

$$\text{RWF} = \sum_{n=1}^{M} \frac{1}{n} \log_{10}(\text{WordFreq}(\mathcal{C}_n^M))$$

A lower RWF indicates less association between token frequencies and top predictions. Results are shown in Figure 4 with M=10. We can see that for most relations, SPE has a lower RWF than baselines, especially P-tuning. These experiments indicate that SPE can mitigate the frequency bias inherently contained in PLMs and avoid answers with spurious associations with the subjects.

C.4 Investigating Percentage of Majority Label in Predictions.

The analyses shown in Appendix C and C.3 focus on relations of type R1 and R2. We now focus on relations of type R3, i.e. relations affected by label imbalance. Results in Figure 5 show that SPE predicts majority training labels less frequently than the baselines in most relations, demonstrating that it is less affected by the imbalances in the label distribution.