The Queen of England is not England's Queen: On the Lack of Factual Coherency in PLMs

Paul Youssef[†] **Jörg Schlötterer**[†‡] **Christin Seifert**[†]
[†]University of Marburg, [‡]University of Mannheim
{paul.youssef, joerg.schloetterer, christin.seifert}@uni-marburg.de

Abstract

Factual knowledge encoded in Pre-trained Language Models (PLMs) enriches their representations and justifies their use as knowledge bases. Previous work has focused on probing PLMs for factual knowledge by measuring how often they can correctly predict an *object* entity given a subject and a relation, and improving fact retrieval by optimizing the prompts used for querying PLMs. In this work, we consider a complementary aspect, namely the coherency of factual knowledge in PLMs, i.e., how often can PLMs predict the *subject* entity given its initial prediction of the object entity. This goes beyond evaluating how much PLMs know, and focuses on the internal state of knowledge inside them. Our results indicate that PLMs have low coherency using manually written, optimized and paraphrased prompts, but including an evidence paragraph leads to substantial improvement. This shows that PLMs fail to model inverse relations and need further enhancements to be able to handle retrieving facts from their parameters in a coherent manner, and to be considered as knowledge bases.

1 Introduction

Pre-trained Language Models (PLMs) are probed for factual knowledge to investigate their usage as knowledge bases, and gain a better understanding of the rich representations they provide (Petroni et al., 2019). Previous extensions have focused on extracting more facts (Zhong et al., 2021; Li et al., 2022b), increasing the consistency of PLMs to paraphrased prompts (Elazar et al., 2021), identifying the parts of PLMs that are responsible for storing knowledge (Dai et al., 2022) and updating facts in them (Meng et al., 2022, 2023).

More recently, Berglund et al. (2023) study the generalization abilities of PLMs from "A is B" to "B is A", and show that if a PLM is trained on "The capital of Malta is Valetta" it will not be able to *correctly* answer the question: "Which country has Valetta as its capital?". In this work, we introduce an intrinsic and complementary aspect, namely the *coherency* of PLMs with respect to factual knowledge. Coherency is not concerned with correctness of the PLMs' predictions, but with the internal state of knowledge in PLMs and its consistency. More concretely, we first ask a PLM to answer the question: "What is the capital of Malta?", and if it answers "Berlin", we ask it to answer the question: "Which country has Berlin as its capital?", and if it answers "Malta", then we say that the PLM has answered coherently (even though the answer is factually wrong). Note that in practice we use Cloze prompts instead of questions to make the task closer to language modeling (see Figure 1). Intuitively, if a human can tell the capital of a country given that country's name, then she is also able to tell the country given its capital's name. Given that PLMs are queried with a subject and a relation to extract a object, we define coherency as the ability of the PLM to infer the subject given its initial prediction for the object entity and vice versa.

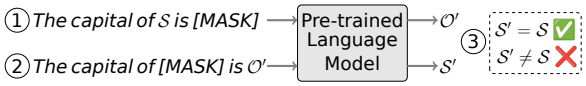


Figure 1: Probing coherency in PLMs. 1) The PLM makes a prediction based on an entity $\mathcal{S}$ and a relation. 2) The PLM makes a second prediction based on the same relation and its first prediction $\mathcal{O}'$. 3) If the PLM predicts $\mathcal{S}$ in the second step it shows coherent behavior.

Our contributions are the following: (1) we introduce coherency to investigate the internal state of factual knowledge in PLMs; (2) we evaluate different PLMs, showing that they have low coherency; (3) we show that optimized and paraphrased prompts do not improve coherency, but the use of evidence paragraphs substantially improves coherency. We make our code available.[1]

[1] https://github.com/paulyoussef/coherency

Findings of the Association for Computational Linguistics: EACL 2024, pages 2342–2354

2 Coherency

PLMs are known to capture vast amount of facts from their pre-training corpora. This has encouraged the community to consider using them as knowledge bases (KBs) (Petroni et al., 2019; Sung et al., 2021), which can be constructed without expensive annotations, and which can easily be queried using natural language. However, the use of PLMs as KBs has many limitations (AlKhamissi et al., 2022). For example, PLMs are quite sensitive to their prompts, and cannot be easily updated with new facts. Factual knowledge in PLMs is estimated by evaluating how often PLMs can correctly predict an object entity $\mathcal{O}$, given a subject entity $\mathcal{S}$ and a relation $\mathcal{R}$, when provided with a prompt which contains the subject and the relation: $t(\mathcal{S},\mathcal{R})$, where t is a function that maps a subject entity and a relation to a prompt in natural language that contains the given entity and expresses the relation in natural language, e.g., `(Malta, capital-of)` $\rightarrow$ `"The capital of Malta is [MASK]"`. In this work, we focus on evaluating the coherency of PLMs with respect to the factual knowledge stored in their parameters, i.e., how often can PLMs predict $\mathcal{S}$ using $t(\mathcal{O}',\mathcal{R})$, given that it predicted $\mathcal{O}'$ using $t(\mathcal{S},\mathcal{R})$. For example, `"The capital of [MASK] is Berlin"` $\rightarrow$ `Malta` is coherent with `"The capital of Malta is [MASK]"` $\rightarrow$ `"Berlin"`. We do not evaluate if the predictions are factually correct, because we are interested in the coherency of the PLMs' world view, regardless of its correctness .We show and discuss correctness scores in Appendix A.

Coherency can be easily calculated for 1-1 relations, but is more challenging, if we consider N-1 or N-M relations, since multiple entities could be correct when trying to predict the subject entity. To address this, we exclude all correct entities except the ground truth subject $\mathcal{S}$ in the second inference step, following Bordes et al. (2013) and Petroni et al. (2019). Since PLMs are known to have certain biases and are sensitive to the prompts, we start with predicting the object given the subject in a first round. In a second round, we start by predicting the subject given the object. The complete algorithm for estimating coherency in PLMs for all types of relations is shown in Algorithm 1. After estimating coherency for each relation, we macro-average over all relations, because we are interested in the average performance for the use case of PLMs-as-KBs, which involves storing facts from different types of relations.

Algorithm 1: Coherency in PLMs

Input: PLM, dataset with n relations
Output: coherency
scores_per_relation = []
`// iterate over relations`
for $i \leftarrow 1$ **to** n **do**
  scores = []
  `// iterate over instances`
  for $j \leftarrow 1$ **to** m **do**
    `// round 1: start with object`
    $\mathcal{O}'_j = PLM(t_i(\mathcal{S}_j,\mathcal{R}_j))$
    exclude correct answers except $\mathcal{S}_j$
    $\mathcal{S}'_j = PLM(t_i(\mathcal{O}'_j,\mathcal{R}_j))$
    if *partial_match*($\mathcal{S}'_j$, $\mathcal{S}_j$) **then**
      scores.append(1)
    else
      scores.append(0)
    `// round 2: start with subject`
    $\mathcal{S}'_j = PLM(t_i(\mathcal{O}_j,\mathcal{R}_j))$
    exclude correct answers except $\mathcal{O}_j$
    $\mathcal{O}'_j = PLM(t_i(\mathcal{S}'_j,\mathcal{R}_j))$
    if *partial_match*($\mathcal{O}'_j$, $\mathcal{O}_j$) **then**
      scores.append(1)
    else
      scores.append(0)
  scores_per_relation.append(mean(scores))
return *mean(scores_per_relation)*

3 Experimental Setup

Here, we describe the data and PLMs, which we use, and our experiments in detail.

3.1 Data

In our experiments, we use the T-REx (Elsahar et al., 2018) subset of LAMA (Petroni et al., 2019), which is often used to estimate factual knowledge in PLMs. T-REx consists of 41 relations with their corresponding templates, and subject-object pairs, for which the relations hold in English. For each of the relations, a manually-written template is provided, which we use to construct the prompts. Some statistics and an example from the T-REx subset are shown in Table 10 in Appendix D.

3.2 How coherent are PLMs?

In this experiment, we aim to find out how coherent are PLMs. We mostly focus on PLMs which are

trained to fill in the blanks based on context, since these make use of a bidirectional context, and we expect them to perform better than autoregressive PLMs on this task. More specifically, we consider BERT (Devlin et al., 2019), InformBERT (Sadeq et al., 2022), T5 (Raffel et al., 2020), and T5-SSM (Guu et al., 2020; Roberts et al., 2020). InformBERT adapts the masking strategy of BERT to focus on more informative tokens. T5-SSM models are additionally trained with Salient Span Masking objective (SSM), which masks only named entities in the pre-training phase. More information about the models are provided in Appendix C. If available, we consider several sizes of the same model in order to investigate the effect of scaling PLMs on coherency. For BERT-based models, we only consider entities that correspond to one token, in order to adhere to the task format from pre-training. We evaluate all models in a zero-shot setting with no finetuning, since we are interested in the coherency of factual knowledge in PLMs after the pre-training phase. For BERT-based models, we choose the token with the highest probability. For T5-models, we use beam search with 10 beams. We use partial match, which returns true if one of the two predictions is contained in the other after converting both to lower case, when comparing the predictions against the ground truth entities.

For completeness, we also evaluate on autoregressive PLMs. More specifically, we consider GPT-2 (Radford et al., 2019) and GPT-Neo (Gao et al., 2020; Black et al., 2021). For autoregressive PLMs, we use typed querying (Kassner et al., 2021), i.e., we extract a probability distribution over a pre-defined set of entities from the model, and choose the most probable entity as the final prediction. Typed querying makes it easy to extract valid answers (entities) from the PLMs' outputs, but also makes the task easier for PLMs since it restricts the output space. We extend the templates from LAMA such that the subject/object entities appear at the very end. We consider autoregressive PLMs only in this experiment.

3.3 Do optimized prompts improve coherency?

Optimizing prompts leads to better fact retrieval (Zhong et al., 2021). In this experiment, we investigate whether optimized prompts lead to higher coherency as well. We utilize Shin et al. (2020)' optimized prompts for T-REx. These prompts differ from one model to another, and from the models we consider, optimized prompts are only available for BERT models.

3.4 Does providing an evidence paragraph increase coherency?

PLMs can fill in the blanks based on the knowledge they have stored in their parameters (parametric knowledge), or based on information that is provided in their inputs (contextual knowledge). The latter boils down to extracting the right information from the input. Previous work has shown that providing evidence paragraphs as additional inputs makes PLMs' predictions more factual (Petroni et al., 2020). Here, we investigate how these evidence paragraphs affect the coherency of factual knowledge in PLMs. The provided evidence paragraphs from LAMA contain a Wikipedia paragraph that expresses the facts. We append the evidence paragraphs to the inputs from the first experiment.

3.5 Is Coherency stable across paraphrased prompts?

PLMs are known to be sensitive to the provided prompts, i.e., small insignificant changes, that preserve the meaning cause the PLMs to change their predictions (Elazar et al., 2021). As a result, retrieving facts from PLMs is highly affected by the prompts used. In this experiment, we consider the effect of using paraphrased prompts on coherency. Does coherency stay the same across different prompts or is it highly variant? We evaluate whether coherency varies with paraphrased prompts from Elazar et al. (2021)'s ParaRel dataset. ParaRel provides paraphrases for 38 of the 41 relations in T-Rex. For each one of the 38 relations, we randomly select a template from ParaRel, and measure how coherency is changed over 10 runs. We consider bert-base and t5-base for this experiment.

4 Results and Discussion

The results for the first three experiments are shown in Table 1. We show the results for autoregressive PLMs separtely in Table 2, because we probe autoregressive PLMs with typed querying. We do not evaluate if the predictions are factually correct. For correctness scores see Table 4 in Appendix A. Since we considered only one-token entities from T-REx for BERT models, we show a normalized version of the results on this subset for better comparability in Table 6, and the results with the total number of instances in Table 7 in Appendix A.

PLMs show poor coherency. We notice that all PLMs have poor coherency. Autoregressive PLMs perform even worse than masked PLMs, even though the task is made easier for them through typed querying (cf. Section 3.2). The poor performance of autoregressive PLMs might be due to their unidirectional training objective, whereas masked PLMs make use of a bidirectional context. Increasing the number of parameters in T5 models leads to consistent improvements in performance. However, this does not generalize to the BERT models (bert-base performs better than bert-large), and to the T5 models that are trained with SSM (t5-large-ssm performs better than t5-3b-ssm). The SSM objective is beneficial for the large variant of T5 (t5-large-ssm improves by 6.5 percentage points over t5-large, and even outperforms t5-3b, which has 4 times as many parameters). Contrarily, this improvement does not generalize to the 3b variant (t5-3b outperforms its SSM counterpart). InformBERT falls short of normal BERT, even though it was shown to outperform BERT, when it comes to facts retrieval (Sadeq et al., 2022). Hence, better facts retrieval does not necessarily affect coherency positively. In general, scaling and entity-centric training objectives have to some extent a positive effect on coherency. We also notice that in most cases models perform worse in the first round. Round 1 can be more difficult, since it may involve predicting a specific subject based on a generic object in the second step (e.g.,"`[MASK] is located in Bern`"), whereas the second round goes into opposite and easier direction ("`University of Bern is located in [MASK]`"). PLMs are known to not provide specific answers (Huang et al., 2023).

We show the results per relation type in Table 5 in Appendix A. The evaluation dataset contains 2 **1-1** relations , 23 **N-1** relations and 16 **N-M** relations with 3 of the 16 **N-M** relations being symmetric. Most PLMs have high coherency on **1-1** relations, but the number of instances for these relations is limited (747 at most), on **N-1**, **N-M** and symmetric relations the performance drops significantly. This shows that **N-1** and **N-M** relations are challenging for PLMs not just with respect to facts retrieval (Petroni et al., 2019), but also with respect to developing a coherent knowledge state.

We also show and categorize examples from different PLMs in Table 8 in Appendix B. In general, one can notice that incoherent predictions are due to : 1) The answer being incorrect in the first step, making it more difficult to predict the answer in the second step (rows 6-7); 2) The templates being not specific enough allowing for non-factual completions (row 8); 3) missing context to retrieve correct relation for non 1-1 relations (row 3).

Optimizing prompts does not help. Optimized prompts lead to a drop in coherency in the second experiment (see results under optimized prompts in Table 1) l. This shows that prompts that better retrieve object entities does not help retrieve the corresponding subject entities. Previous work showed that optimized prompts overfit the facts distribution of objects (Cao et al., 2021), which might negatively affect their ability to retrieve the subject entities. This is also evident by the difference in scores between the two rounds.

Evidence paragraphs improve coherency. Including evidence paragraphs in the inputs substantially improves performance (see results under evidence paragraphs in Table 1). This shows that PLMs are better at extracting answers from their inputs than recalling them from their parameters. In fact, adding an evidence paragraph reduces the performance gaps among models of different sizes and pre-training objectives. This suggests that retrieval-based approaches are indeed a promising alternative to scaling language models (Kandpal et al., 2023). Still, coherency is not high under this setting as well. We believe this is due to the PLMs failing to extract the correct entities or to the conflicts between contextual and parametric knowledge in PLMs (Neeman et al., 2023).

Coherency varies across paraphrases. Table 3 shows the minimum, average and maximum coherency scores with paraphrased prompts. A breakdown in relations is available in Appendix A (Fig. 2).[2] As with fact retrieval, the results indicate that prompts have a significant effect on the performance. For example, there are more than 25 percentage points difference in coherency between the min and max scores for t5-base. Still, even when considering the best prompts, the overall coherency score is low.

In general, our analysis shows that PLMs do not possess a coherent knowledge state. The low coherency might be due: 1) The fact that PLMs make predictions based on shallow surface level features (Poerner et al., 2020; Li et al., 2022a), which makes PLMs output relevant but incoherent

[2]Note that, for this experiment, we use only 38 of the 41 relations in T-Rex – The ones for which paraphrases exist.

and non-factual predictions (for an example see row 6 in Table 8). This is inherent to all PLMs, and requires further architectural improvements; 2) The training data for PLMs, which might be biased towards certain entities (the more frequent ones); 3) The uni-directional training in the case of autoregressive PLMs that makes PLMs sensitive to the order in which the entities are observed.

Model	Round 1	Round 2	Avg.
bert-base-uncased	9.74	11.81	10.78
bert-large-uncased	9.83	10.29	10.06
InformBERT	8.04	11.55	9.79
t5-base	9.02	10.29	9.66
t5-large	9.07	12.03	10.55
t5-3b	8.62	23.90	16.26
t5-large-ssm	**9.89**	**24.23**	**17.06**
t5-3b-ssm	8.97	20.88	14.92
w/ optimized prompts			
bert-base-uncased	1.52	**12.80**	**7.16**
bert-large-uncased	**1.87**	7.38	4.62
w/ evidence paragraphs			
bert-base-uncased	22.30	39.87	31.09
bert-large-uncased	21.05	41.98	31.52
InformBERT	43.07	46.40	44.74
t5-base	41.40	58.31	**49.85**
t5-large	31.46	55.15	43.31
t5-3b	27.06	**62.89**	44.98
t5-large-ssm	**50.17**	43.97	47.07
t5-3b-ssm	48.52	41.81	45.17

Table 1: Coherency score per round and on average for different PLMs using manually-written, optimized prompts and evidence paragraphs. The highest performance under each category is in **bold**, and the best performance overall is underlined.

5 Related Work

Reversal curse. Berglund et al. (2023) investigate the generalization abilities of autoregressive PLMs from one data form, that is encountered during training (A is B), to another (B is A), showing a generalization failure. Berglund et al. (2023) refer to this generalization inability in autoregressive PLMs as the *reversal curse*. Our work is close but complementary to theirs. We focus on the coherency of the internal state of factual knowledge in autoregressive *and* masked PLMs, *regardless* of how correct the PLMs' predictions are.

Model	Round 1	Round 2	Avg.
gpt2	0.24	3.98	2.11
gpt-neo-1.3B	0.44	**12.85**	**6.65**
gpt-neo-2.7B	**0.56**	11.82	6.19

Table 2: Coherency score per round and on average for autoregressive PLMs using manually-written prompts. The highest performance is in **bold**. Autoregressive PLMs are probed using typed querying.

Model	Min.	Avg.	Max.	#Instances
bert-base-uncased	3.74	11.16	19.25	2852
t5-base	6.51	16.88	31.69	27788

Table 3: Coherency scores with different paraphrases. We show the results with the worst/average/best performing prompts per relation.

Factual knowledge in PLMs. PLMs contain vast amounts of linguistic (Tenney et al., 2019; Jawahar et al., 2019), commonsense (Davison et al., 2019) and factual knowledge (Roberts et al., 2020) that is captured during pre-training. Many works focus on factual knowledge in PLMs (Youssef et al., 2023), since factual knowledge is said to contribute to the rich presentations produced by PLMs, and potentially justifies the use of PLMs as KBs (Petroni et al., 2019; Ye et al., 2022). For example, Shin et al. (2020); Zhong et al. (2021) optimize prompts to extract more facts from PLMs, Elazar et al. (2021); Fierro and Søgaard (2022) investigate the sensitivity of PLMs to paraphrased prompts, (Malkin et al., 2022; Wang et al., 2023) debias the outputs of PLMs for better facts extraction, Meng et al. (2022, 2023) address editing facts in PLMs to make it possible to correct and update facts. However, these works collectively focus on extrinsic aspects. We focus on a more intrinsic aspect, i.e., the coherency of factual knowledge inside PLMs. This complements aspects addressed in previous work.

6 Conclusion

In this work, we focused on evaluating the coherency of factual knowledge in PLMs. We considered the use of manually-written, optimized, and paraphrased prompts. Our results indicate poor coherency. The inclusion of an evidence paragraph leads to substantial improvements. This shows that PLMs can leverage contextual knowledge better than parametric knowledge and highlights the importance of retrieval-augmented PLMs. We believe that further improvements are needed to improve coherency in PLMs, and to consider them as alternatives to KBs. We believe that future work should focus on further improving PLMs on the architectural level, the data level, and the interface between them (pre-training objectives).

7 Limitations

Coherency can be easily determined using 1-1 relations. For N-1 or N-M relations, some potential answers should be excluded. However, it is quite difficult to exclude every possible answer for certain relations (e.g., everyone who is an English native speaker) from the model's vocabulary. We only excluded answers that are present in LAMA, following previous work (Bordes et al., 2013) and (Petroni et al., 2019). This might have had a negative effect on the results (cf. Section 4, discussion of lower scores in round 1).

Acknowledgement

We thank Alessandro Noli for helpful discussions.

References

Badr AlKhamissi, Millicent Li, Asli Celikyilmaz, Mona Diab, and Marjan Ghazvininejad. 2022. A review on language models as knowledge bases. *arXiv preprint arXiv:2204.06031*.

Lukas Berglund, Meg Tong, Max Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. 2023. The reversal curse: Llms trained on" a is b" fail to learn" b is a". *arXiv preprint arXiv:2309.12288*.

Sid Black, Gao Leo, Phil Wang, Connor Leahy, and Stella Biderman. 2021. GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow. If you use this software, please cite it using these metadata.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

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.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502, Dublin, Ireland. Association for Computational Linguistics.

Joe Davison, Joshua Feldman, and Alexander Rush. 2019. Commonsense knowledge mining from pretrained models. 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 1173–1178, Hong Kong, China. Association for Computational Linguistics.

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

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. 2021. Measuring and improving consistency in pretrained language models. *Transactions of the Association for Computational Linguistics*, 9:1012–1031.

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)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Constanza Fierro and Anders Søgaard. 2022. Factual consistency of multilingual pretrained language models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 3046–3052, Dublin, Ireland. Association for Computational Linguistics.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. ICML'20. JMLR.org.

Jie Huang, Kevin Chen-Chuan Chang, Jinjun Xiong, and Wen-mei Hwu. 2023. Can language models be specific? how? In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 716–727, Toronto, Canada. Association for Computational Linguistics.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2023. Large language models struggle to learn long-tail knowledge. In *International Conference on Machine Learning*, pages 15696–15707. PMLR.

Nora Kassner, Philipp Dufter, and Hinrich Schütze. 2021. Multilingual LAMA: Investigating knowledge in multilingual pretrained language models. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3250–3258, Online. Association for Computational Linguistics.

Shaobo Li, Xiaoguang Li, Lifeng Shang, Zhenhua Dong, Chengjie Sun, Bingquan Liu, Zhenzhou Ji, Xin Jiang, and Qun Liu. 2022a. How pre-trained language models capture factual knowledge? a causal-inspired analysis. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1720–1732, Dublin, Ireland. Association for Computational Linguistics.

Yiyuan Li, Tong Che, Yezhen Wang, Zhengbao Jiang, Caiming Xiong, and Snigdha Chaturvedi. 2022b. SPE: Symmetrical prompt enhancement for fact probing. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11689–11698, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Nikolay Malkin, Zhen Wang, and Nebojsa Jojic. 2022. Coherence boosting: When your pretrained language model is not paying enough attention. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8214–8236, Dublin, Ireland. Association for Computational Linguistics.

Kevin Meng, David Bau, Alex J Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*.

Kevin Meng, Arnab Sen Sharma, Alex J Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations*.

Ella Neeman, Roee Aharoni, Or Honovich, Leshem Choshen, Idan Szpektor, and Omri Abend. 2023. DisentQA: Disentangling parametric and contextual knowledge with counterfactual question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10056–10070, Toronto, Canada. Association for Computational Linguistics.

Fabio Petroni, Patrick Lewis, Aleksandra Piktus, Tim Rocktäschel, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. 2020. How context affects language models' factual predictions.

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.

Nina Poerner, Ulli Waltinger, and Hinrich Schütze. 2020. E-BERT: Efficient-yet-effective entity embeddings for BERT. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 803–818, Online. Association for Computational Linguistics.

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

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.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. 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.

Mujeen Sung, Jinhyuk Lee, Sean Yi, Minji Jeon, Sungdong Kim, and Jaewoo Kang. 2021. Can language models be biomedical knowledge bases? In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4723–4734, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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*.

Yuhang Wang, Dongyuan Lu, Chao Kong, and Jitao Sang. 2023. Towards alleviating the object bias in

prompt tuning-based factual knowledge extraction. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4420–4432, Toronto, Canada. Association for Computational Linguistics.

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022. ZeroGen: Efficient zero-shot learning via dataset generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11653–11669, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Paul Youssef, Osman Koraş, Meijie Li, Jörg Schlötterer, and Christin Seifert. 2023. Give me the facts! a survey on factual knowledge probing in pre-trained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15588–15605, Singapore. 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 Results

Correctness. We investigate how correct the PLMs' predictions are. For each instance, we count how often the first prediction in the first round (**c1**), and in the second round (**c2**) were correct. We only consider the first predictions in each round, since having the incorrect answer in the first inference step of each round makes it more difficult for the model to answer correctly in the second inference step. We also report how often are all predictions correct (**all correct**). We calculate each score for all relations, and average over all relations. Results are shown in Table 4. We notice that for all models the **c1** scores are higher than the **c2** scores. We believe this is because in the first inference step in round 1, models predict *object* entities, whereas in the first step of round 2 they predict *subject* entities. Predicting subject entities is more difficult, since their corresponding mask tokens are placed at the beginning of the templates. This allows for valid completions that do not contain any entities. For example, if the template is "`[MASK] is the capital of Malta`", then "`It`" is also a valid completion with no entities. Additionally, predicting the subject entity based on the object entity might be ambiguous (see discussion in Section 4).

Coherency scores per relation type. Coherency scores per relation type are shown in Table 5.

Coherency on a subset. Table 6 shows a normalized version of the coherency scores using manually-written prompts.

Coherency over relations with different paraphrases. Figure 2 shows the average coherency scores with standard deviation over different relations when using paraphrased prompts. Note that bert-base-uncased has less relations than t5-base (36 vs. 38), since some relations ended up with no instances after excluding multi-token entities. In general, we notice high standard deviation for most relations.

Coherency scores with the number of instances. Table 7 shows the coherency scores with the size of the test set in instances.

B Examples

We show examples of several failures from different prompts and categorize these in Table 8.

C Additional Details on Masked Language Models

Masked PLMs are trained to predict one or several tokens given a context. This is considered a generalization of the conventional language modeling objective that predicts the next token based on its left context. BERT (Devlin et al., 2019), an encoder-only model, was trained using the Maksed Language Modeling (MLM) objective. T5, an encoder-decoder model, was also trained using a variant of the MLM objective in addition to a mixture of supervised tasks. In the Salient Span Masking (SSM) versions of T5, the models are additionally trained by masking only entities to push the model to focus more on these (Guu et al., 2020; Roberts et al., 2020). Similarly, Sadeq et al. (2022) leverage pointwise mutual information to mask salient tokens in an unsupervised manner. Table 9 provides an overview of the architecture and the number of parameters for each model.

D Choice of Datasets

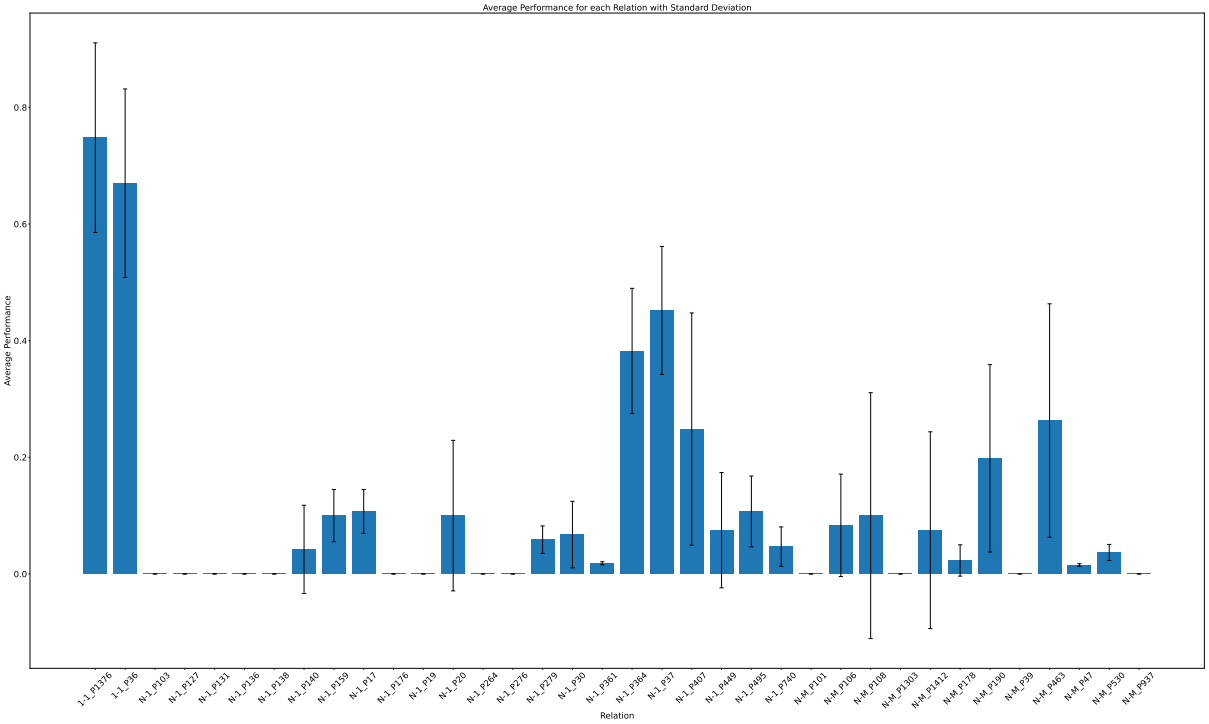
The LAMA probe (Petroni et al., 2019) has been proposed to assess how much factual knowledge is contained in PLMs. We believe it is suitable for the experiments we conduct, since it consists of (subject, relation, object) triples. This allows

Model	c1	c2	All correct	#relations	#Instances
bert-base-uncased	30.77	8.55	4.27	39	2919
bert-large-uncased	25.96	8.39	4.22	39	2919
InformBERT	22.33	5.97	4.34	39	2926
t5-base	11.03	6.21	1.30	41	29672
t5-large	14.77	6.26	1.70	41	29672
t5-3b	20.93	6.10	2.33	41	29672
t5-large-ssm	18.42	4.69	2.73	41	29672
t5-3b-ssm	19.61	4.28	2.96	41	29672
Autoregressive PLMs					
gpt2	7.70	0.43	0.04	41	29672
gpt-neo-1.3B	17.65	0.93	0.13	41	29672
gpt-neo-2.7B	18.50	1.31	0.22	41	29672
w/ optimized prompts					
bert-base-uncased	25.27	1.49	0.02	39	2919
bert-large-uncased	31.92	2.94	0.10	39	2919
w/ evidence paragraphs					
bert-base-uncased	46.98	19.97	11.12	39	2919
bert-large-uncased	49.66	20.27	12.98	39	2919
InformBERT	49.42	45.92	24.95	39	2926
t5-base	59.77	39.28	23.99	41	29672
t5-large	59.31	27.57	15.77	41	29672
t5-3b	57.35	23.17	11.73	41	29672
t5-large-ssm	44.47	47.61	23.10	41	29672
t5-3b-ssm	41.44	46.40	21.41	41	29672

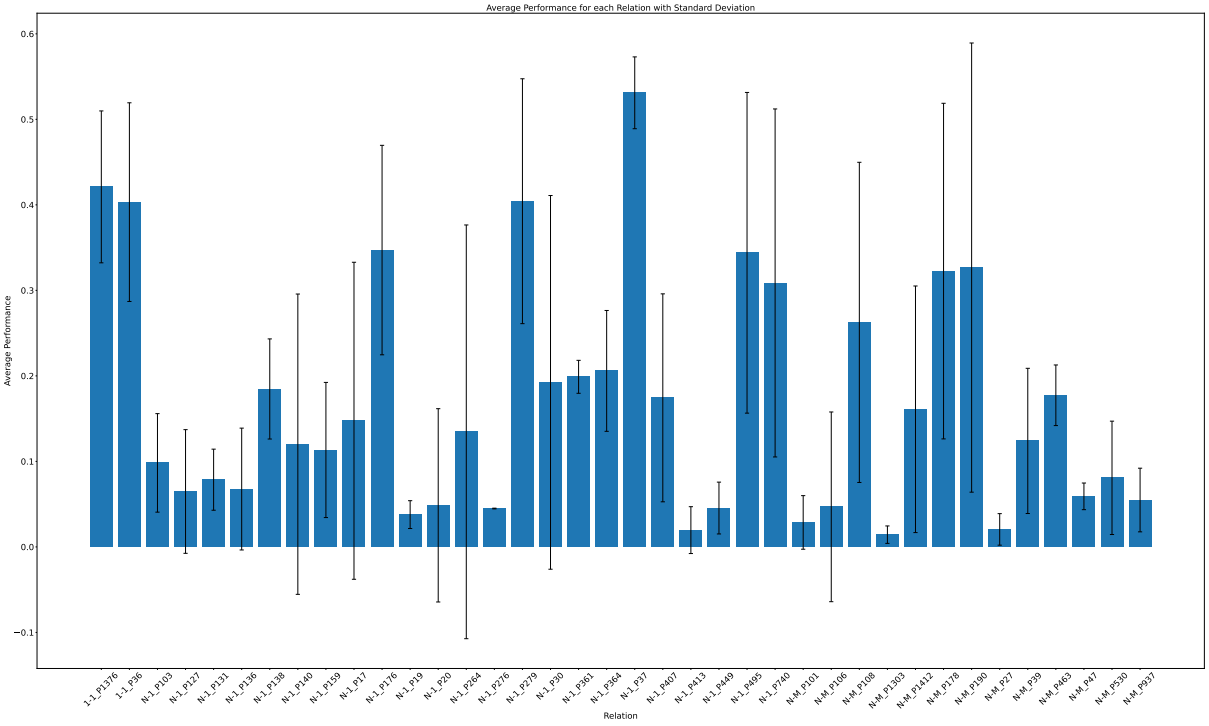
Table 4: Correctness scores in the first inference step of the first round (**c1**), the second round (**c2**), and in all inference steps (**all correct**). Results are averaged over all relations. BERT-based models have less relations and instances, because we consider only one-token entities for these models.

Relation Type	1-1		N-1		N-M		symmetric		All	
Model	Coherency	#Instances	Coherency	#Instances	Coherency	#Instances	Coherency	#Instances	Coherency	#Instances
bert-base-uncased	84.11	232	5.93	633	8.10	2054	12.57	1927	10.78	2919
bert-large-uncased	82.71	232	6.65	633	5.38	2054	15.10	1927	10.06	2919
InformBERT	81.03	232	5.28	637	6.91	2057	18.46	1929	9.79	2926
t5-base	36.84	747	8.55	16838	7.84	12087	8.61	2882	9.66	29672
t5-large	48.90	747	6.90	16838	11.02	12087	14.87	2882	10.55	29672
t5-3b	61.21	747	14.84	16838	12.68	12087	21.41	2882	16.26	29672
t5-large-ssm	75.96	747	17.22	16838	9.46	12087	7.44	2882	17.06	29672
t5-3b-ssm	76.36	747	13.94	16838	8.66	12087	13.00	2882	14.92	29672
Autoregressive PLMs										
gpt2	0.26	747	1.46	16838	3.27	12087	0.16	2882	2.11	29672
gpt-neo-1.3B	3.40	747	9.71	16838	2.65	12087	0.19	2882	6.65	29672
gpt-neo-2.7B	4.59	747	6.37	16838	6.14	12087	0.51	2882	6.19	29672
w/ optimized prompts										
bert-base-uncased	1.46	232	7.54	633	7.35	2054	2.36	1927	7.16	2919
bert-large-uncased	2.38	232	6.85	633	1.66	2054	7.23	1927	4.62	2919
w/ evidence paragraphs										
bert-base-uncased	87.78	232	26.49	633	30.27	2054	22.66	1927	31.09	2919
bert-large-uncased	90.30	232	28.13	633	28.65	2054	26.61	1927	31.52	2919
InformBERT	84.06	232	42.05	637	43.43	2057	31.86	1929	44.74	2926
t5-large-ssm	84.73	747	46.38	16838	43.35	12087	32.50	2882	47.07	29672
t5-3b-ssm	86.95	747	44.80	16838	40.47	12087	27.10	2882	45.17	29672
t5-base	73.86	747	49.91	16838	46.77	12087	30.37	2882	49.85	29672
t5-large	66.94	747	42.65	16838	41.30	12087	26.10	2882	43.31	29672
t5-3b	74.16	747	45.32	16838	40.84	12087	26.93	2882	44.98	29672

Table 5: Coherency scores per relation type.



(a) bert-base-uncased



(b) t5-base

Figure 2: Average coherency with standard deviation when using paraphrased prompts over different relations.

Model	Coherency	#Instances
bert-base-uncased	10.78	2919
bert-large-uncased	10.06	2919
InformBERT	9.79	2919
t5-base	10.64	2919
t5-large	12.45	2919
t5-3b	17.39	2919
t5-large-ssm	16.76	2919
t5-3b-ssm	14.57	2919
Autoregressive PLMs		
gpt2	2.36	2919
gpt-neo-1.3B	4.89	2919
gpt-neo-2.7B	11.50	2919

Table 6: Coherency of different PLMs on a subset of one-token entities using BERT's tokenizer with manually-written prompts.

us to evaluate, how often PLMs can predict one entity (either the subject or object) given the other entity and the relation. Additionally, LAMA covers 41 relations of different types, which helps us provide a coherency estimate based on all of these relations. See Table 10 for an overview. We also used the ParaRel dataset (Elazar et al., 2021). This dataset has been proposed to measure the sensitivity of PLMs to paraphrased prompts with respect to factual knowledge. Similarly, we use ParaRel to investigate how the coherency score is affected by paraphrased prompts. All the datasets we used are in English. Additionally, we used the prompts obtained by Autoprompt (Shin et al., 2020) to investigate the effect of having optimized prompts on the performance. We manually create prompts for autoregressive PLMs. These templates are included with our code. [3]

E Computational Resources

In all of our experiments, we use a NVIDIA A100 GPU with 80GB of memory. Our experiments took roughly 25 GPU days.

Model	Coherency	#Instances
bert-base	10.78	2919
bert-large	10.06	2919
InformBERT	9.79	2926
t5-base	9.66	29672
t5-large	10.55	29672
t5-3b	16.26	29672
t5-large-ssm	**17.06**	29672
t5-3b-ssm	14.92	29672
Autoregressive PLMs		
gpt2	2.11	29672
gpt-neo-1.3B	6.65	29672
gpt-neo-2.7B	6.19	29672
w/ optimized prompts		
bert-base	**7.16**	2919
bert-large	4.62	2919
w/ evidence paragraphs		
bert-base-uncased	31.09	2919
bert-large-uncased	31.52	2919
InformBERT	44.74	2926
t5-base	**49.85**	29672
t5-large	43.31	29672
t5-3b	44.98	29672
t5-large-ssm	46.78	29482
t5-3b-ssm	45.17	29672

Table 7: Coherency for different PLMs using manually-written, optimized prompts and evidence paragraphs. The highest performance under each category is in **bold**, and the best performance overall is underlined.

[3] https://github.com/paulyoussef/coherency

Type	Model	Relation	Forward	Backward	ID
Coherent & Correct	bert-base-uncased	edmonton, alberta	edmonton is the capital of [MASK] → alberta	[MASK] is the capital of alberta → edmonton	1
Coherent & Incorrect	t5-large	Brunei, Malay	The official language of Brunei is [MASK] → Bruneian	The official language of [MASK] is Bruneian → Brunei	2
Incoherent & Correct (1st)	bert-base-uncased	lucknow, urdu	The official language of lucknow is [MASK] → urdu	The official language of [MASK] is urdu → maldives	3
	gpt-neo 2.7B	Topeka, Kansas	Topeka is the capital of [MASK] → Kansas	Kansas's capital is [MASK] → Quebec City	4
Repetition	informBERT	iPhone, Apple	iPhone is produced by [MASK] → apple	[MASK] is produced by apple → apple	5
Incoherent & Incorrect	bert-large-uncased	lille, nord	lille is the capital of [MASK] → france	[MASK] is the capital of france → lyon	6
Repetition	t5-base	Germany, Berlin	The capital of Germany is [MASK] → Frankfurt am Main	The capital of [MASK] is Frankfurt am Main → Frankfurt am Main	7
Pronoun	bert-base-uncased	munich, germany	munich is located in [MASK] → bavaria	[MASK] is located in bavaria → it	8

Table 8: Examples from different PLMs.

Model	#Parameters	Architecture
bert-base	110M	encoder-only
bert-large	345M	encoder-only
InformBERT	110M	encoder-only
t5-base	220M	encoder-decoder
t5-large	770M	encoder-decoder
t5-3B	3B	encoder-decoder
t5-11B	11B	encoder-decoder
gpt-2	117M	decoder-only
gpt-neo 1.3B	1.3B	decoder-only
gpt-neo 2.7B	2.7B	decoder-only

Table 9: Models with number of parameters and architectures. SSM variants of t5 have the same number of parameters as their normal counterparts.

#Relations	#Instances	Example
41	29672	`X was born in Y`

Table 10: Statistics of LAMA and an example.

Dataset	License
LAMA	CC-BY-NC 4.0
ParaRel	MIT License
Optimized prompts	Apache License 2.0

Table 11: Licenses of the datasets used in this work.