

Improving Language Model Predictions via Prompts Enriched with Knowledge Graphs^{*}

Ryan Brate¹, Minh-Hoang Dang², Fabian Hoppe^{3,4}, Yuan He⁵,
Albert Meroño-Peñuela⁶ and Vijay Sadashivaiah⁷

¹*KNAW Humanities Cluster, Digital Humanities Lab, Amsterdam, Netherlands*

²*LS2N, Université de Nantes, faculté des Sciences et Techniques (FST), France*

³*FIZ Karlsruhe, Leibniz Institute for Information Infrastructure, Germany*

⁴*Karlsruhe Institute of Technology, Institute AIFB, Germany*

⁵*University of Oxford, UK*

⁶*King's College London, UK*

⁷*Rensselaer Polytechnic Institute, USA*

Abstract

Despite advances in deep learning and knowledge graphs (KGs), using language models for natural language understanding and question answering remains a challenging task. Pre-trained language models (PLMs) have shown to be able to leverage contextual information, to complete cloze prompts, next sentence completion and question answering tasks in various domains. Unlike structured data querying in e.g. KGs, mapping an input question to data that may or may not be stored by the language model is not a simple task. Recent studies have highlighted the improvements that can be made to the quality of information retrieved from PLMs by performing amendments to otherwise naive prompts. In this paper, we explore the effects of enriching prompts with additional contextual information leveraged from the Wikidata KG on language model performance. Specifically, we compare the performance of naive vs. KG-engineered cloze prompts for genre completion in the movie domain. Selecting a broad range of commonly available Wikidata properties, we show that enrichment of cloze-style prompts with Wikidata information can result in a significantly higher recall in the selected BERT and RoBERTa large PLMs. However, it is also apparent that the optimum level of data enrichment differs between models.

Keywords

Prompt Learning, Pre-trained Language Model, Knowledge Graph.

1. Introduction

Pre-trained language models (PLMs) [1, 2], based on deep learning attention-based architectures, have shown to have outstanding performance at various natural language processing (NLP) tasks predicated on natural language understanding. However, the extent to which they capture domain knowledge and *empirical semantics* [3] — i.e. the use of formal domain properties in practice — is not well understood. In this work, we narrow down the focus to cloze-style


Workshop on Deep Learning for Knowledge Graphs (DL4KG@ISWC2022), October 23-24, 2022

^{*} Authors listed in alphabetical order

✉ r.brates@gmail.com (R. Brate); minhhoangdang@hotmail.com (M. Dang); fabian.hoppe@kit.edu (F. Hoppe); yuan.he@cs.ox.ac.uk (Y. He); albert.merono@kcl.ac.uk (A. Meroño-Peñuela); sadasv2@rpi.edu (V. Sadashivaiah)

🆔 0000-0002-7047-2770 (F. Hoppe); 0000-0002-4486-1262 (Y. He); 0000-0003-3375-3810 (V. Sadashivaiah)

© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

question answering, the task of predicting the masked entity text in a sentence. For example, given: “The Klingons are a species in the franchise [MASK]”, the PLM is expected to predict “Star Trek” for [MASK]. It aims to extract the implicit knowledge entailed by the PLMs, since such knowledge can be used for downstream NLP applications like sentiment analysis [4], dialogue systems [5], and natural language inference [6], as well as for completing the missing information of knowledge graphs or ontologies [7], and even constructing new ones [8].

In recent years, PLMs have improved on the state of the art in many NLP tasks by leveraging large text corpora [9], but mostly requiring annotated data for task-specific fine-tuning [10]. However, the empirical semantics gathered by these models is limited to distributional aspects [11]. Therefore, the performance, especially in the few- and zero-shot setting, highly depends on the provided *prompt*, i.e. snippets of contextual information for a specific task. However, in many cases the engineering of the prompts is naive and simplistic, giving the PLM too little context to provide an accurate answer; and unsystematic, providing little principles on how exactly these prompts need to be composed in order to have a predictable behaviour. Indeed, recent studies [12] have highlighted the improvements that can be made to the quality of information retrieved from PLMs by performing amendments to these prompts. This casts doubts on some studies [13] that claim that a PLM cannot answer easy questions about e.g. culture (movies, books, music); it is reasonable to postulate that PLMs could perhaps answer those questions accurately if they were provided with systematically engineered prompts that contained richer contexts.

Existing approaches of prompt engineering include: (i) learn-by-example, where the prompt consists of the concatenation of correct examples we expect a PLM to predict [2]; (ii) manually designed prompts of different granularities [13]; (iii) automatically searched prompts optimized on few-shot samples [14], all of which rely on the implicit semantics of natural language texts. In this paper, we investigate how incorporating explicit knowledge from external sources like knowledge graphs can help prompt engineering and thus enhance the cloze-style question answering of PLMs. Specifically, we explore cloze-style prompts with respect to the movie domain in respect of the performance of the BERT and RoBERTa large PLMs.

2. Related Work

Studies towards prompt learning are based on the hypothesis that the pre-trained language models (PLMs) have learnt abundant knowledge but just require sufficiently detailed contexts for prediction [2, 10, 15] – and in this way, we can apply PLMs without data-driven fine-tuning. A (hard¹) *prompt* is the conditioning text coming with the input text for providing contexts or hints to PLMs. A *template* (a.k.a. *pattern*) is a function that integrates the inputs and prompts. Answers are then given by the PLMs conditioned on the prompts, and a further function (i.e., *verbalizer*) is often required to map the answers to the final outputs. The reason for that is, the prompt learning paradigm is typically formulated as a similar task to the PLM’s pre-training task, which does not necessarily yield the desired outputs of downstream applications.

An important part of prompt learning is prompt engineering, i.e., to design template(s), either manually or automatically, to support downstream applications. In [2], Brown et al. proposed

¹Soft prompts are learnt at the embedding level.

to use *demonstrations*, i.e., a sequence of input-output texts, as the prompts, expecting that the PLM can implicitly learn to predict from examples. For instance, if we want the PLM to predict the masked position in “[MASK] is the capital of China.”, we can demonstrate by appending “London is the capital of the UK” after the masked sentence. Schick et al. [16] manually designed different templates, each corresponding to an individual PLM trained on few-shot examples. The predictions of downstream text classification and natural language inference tasks were then made according to the ensemble of trained PLMs. The work of Shin et al. [14] argued that manually designed templates suffer from the uncertainty of guesswork or the lack of domain expertise. Therefore, they proposed to search for templates using gradient-based optimization. More recently, Lu et al. [17] have shown that PLMs’s performance varies with the order of these prompts, and use generative LMs and entropy statistics on the prompt permutations to identify prompts with good performance.

Knowledge graphs or ontologies are excellent sources for providing explicit knowledge to enrich the prompts or the verbalizers. West et al. [18] considered *distilling* a student model in the common sense domain from the enormously large PLM, GPT-3 [2], which serves as the teacher model. They adopted the prompt learning scheme to extract triples from the teacher model with templates created and examples extracted from the common sense knowledge graph Atomic [19]. Hu et al. [7] argued that the label word space (i.e., the answer space) can be well expanded by adding in external knowledge about related words. They then employed different refinement heuristics to shortlist candidates to benefit the downstream classification task. For instance, if some “Person” is classified as a “Physicist” in the ground truth data, then answers like “Scientist” will also be accepted.

Our work was motivated by the probing study of Penha et al. [13] that investigates whether BERT (a well-known PLM consisting of the stacked transformer encoders [1]) actually knows superficial cultural knowledge about books, movies, and music. Cloze-style questions for classifying the genre of entities (from Wikidata) of different books, movies, and music were given for the PLM to answer, often with unsatisfying performance. However, their work considered naive prompts without sufficient contexts, while ours attempts to examine if knowledge graphs can enrich these prompts, especially giving additional contexts (e.g., attributes, k -hop neighbours) of the entities in order to help the PLM to predict better.

3. Methodology

3.1. Prompt Engineering

Similarly to [13], we consider an entity genre classification task. The prompts are of the form: “<title> is a movie <Wikidata enrichment>, of the genre [MASK].”, Where <Wikidata enrichment> is an aggregation of movie properties and corresponding values extracted from Wikidata pertaining the title in question, in some natural language format. Table 1 lists the Wikidata properties used to assemble values for <Wikidata enrichment>.

The Wikidata properties listed in Table 1 are broadly ranked in descending information specificity. It was in this order, that 10 variations for a probe were constructed, by sequentially adding Wikidata properties to prompts, building gradually more contextual-information dense prompts. In adding property information, only the first value of each Wikidata property was

Wikidata property	Property Label	Enrichment Text
wdt:P161	cast member	starring
wdt:P57	director	directed by
wdt:P162	producer by	produced by
wdt:P58	screenwriter	screenwriter
wdt:P86	composer	music by
wdt:P1040	film editor	edited by
wdt:P577	year	released
wdt:P750	distributed by	distributed by
wdt:P495	country of origin	originating from

Table 1

Wikidata properties used in constructing probes for the movie dataset. 'enrichment text' is the text adopted in the probe enrichment to describe the property in question in a more natural language format.

used where more than one was available (e.g., the first listed cast member). E.g., as follows; the unenriched prompt, the first 2 successive prompt enrichments, and the final enriched form pertaining to the movie Die Hard.:

- non-enriched prompt: Die Hard is a movie, of the genre [MASK].
- enriched Prompt 1(A): Die Hard is a movie, *starring Bruce Willis*, of the genre [MASK].
- enriched Prompt 2(A): Die Hard is a movie, *starring Bruce Willis, directed by John McTiernan*, of the genre [MASK].
- enriched Prompt 9(A): Die Hard is a movie, *starring Bruce Willis, directed by John McTiernan, produced by Joel Silver, screenwriter Roderick Thorp, music by Michael Kamen, edited by John F. Link, released 1988, distributed by Netflix, originating from United States of America*, of the genre [MASK].

Given the potential for sensitivity of PLMs to the verbalisation strategy used in the construction of cloze-type prompts. We considered 2 verbalisation strategies for aggregation of the additional Wikidata properties. Whereas the above *verbalisation strategy A* form is aggregated with commas, the *verbalisation strategy B* form is aggregated with *and* tokens. E.g.:

- enriched Prompt 1(B): Die Hard is a movie *and starring Bruce Willis*, of the genre [MASK].
- enriched Prompt 2(B): Die Hard is a movie *and starring Bruce Willis and directed by John McTiernan*, of the genre [MASK].
- enriched Prompt 9(B): Die Hard is a movie *and starring Bruce Willis and directed by John McTiernan and produced by Joel Silver and screenwriter Roderick Thorp and music by Michael Kamen and edited by John F. Link, released 1988 and distributed by Netflix and originating from United States of America*, of the genre [MASK].

Thus, in total 19 prompt variations are considered for each movie.

3.2. Knowledge Graph Querying

The auxiliary data for each movie is extracted from Wikidata. This is done in a simplistic two step process using SPARQL queries. The queries operate on a batch of input records to reduce the number of requests and avoid timeout errors.

First, the movies are linked to their respected Wikidata entities by IMDb or TMDb ID utilizing the Wikidata properties *IMDb ID* (wdt:P345) and *TMDb movie ID* (wdt:P4947). If this does not provide an entity an exact string matching given the title is attempted as well.

```
SELECT ?mId ?imdbId ?tmdbId ?movie
WHERE {
  VALUES (?mId ?imdbId ?tmdbId) { ("1" "tt0114709" "862" ) ... }
  {?movie wdt:P345 ?imdbId . }
  UNION
  {?movie wdt:P4947 ?tmdbId . }
}
```

Listing 1: SPARQL query used for entity linking with the IMDb or TMDb ID.

The second step queries the entities for the auxiliary data used to enrich the prompts with additional contextual information. Therefore, a set of 10 manually selected domain-specific properties are extracted for each entity. The properties are selected based on human intuition and the most frequent co-occurrence for the given entities. Overall, a set of 28 properties was investigated. A simplified version of the utilized SPARQL query is given in 2. This query can easily be adapted to query other properties by adding these properties to the *?property* values.

```
SELECT ?mId (SAMPLE(?movieLabel) AS ?movieLabel) (SAMPLE(?propertyLabel) AS ?propertyLabel)
(GROUP_CONCAT(DISTINCT ?objectLabel; SEPARATOR=", ") AS ?objectList)
WHERE{
  VALUES (?mId ?movie) { ("1" ) ...}
  ?movie rdfs:label ?movieLabel .
  FILTER (LANG(?movieLabel)="en")
  VALUES ?property {wdt:P144 wdt:P179 ...}
  ?p1 wikibase:directClaim ?property .
  ?p1 rdfs:label ?propertyLabel .
  FILTER (LANG(?propertyLabel)="en")
  OPTIONAL {
    ?movie ?property ?object .
    ?object rdfs:label ?objectLabel .
    FILTER (LANG(?objectLabel)="en")
  }
}
GROUP BY ?mId ?property
```

Listing 2: Simplified SPARQL query used to retrieve additional movie knowledge from Wikidata.

3.3. Language Model Probing

The basic idea of our method is to use the information about entities in knowledge graphs to expand cloze-style prompts with richer entity descriptions, as shown in Figure 1. For example, we enrich the naive prompt *Die Hard* (1988) is of genre <mask> for a QA task on movie genres through (a) matching the Wikidata item for the movie *Die Hard* (1988) with the query in Listing 1, and then we use the 28 property values returned by the query in Listing 2. We use property literals and verbalize entities to compose valid phrases. As a result, we obtain e.g. *Die hard* (1988 film about hostage crisis starring Bruce Willis) is of genre <mask>.

We then use both (a) the naive prompts and (b) the knowledge graph-enriched prompts to query various language models, and compare their performance at QA tasks such as entity genre classification. Our assumption is that systematically informed and semantically richer prompts will show higher performance at QA answering tasks than naive prompts. We investigate this in the next sections.

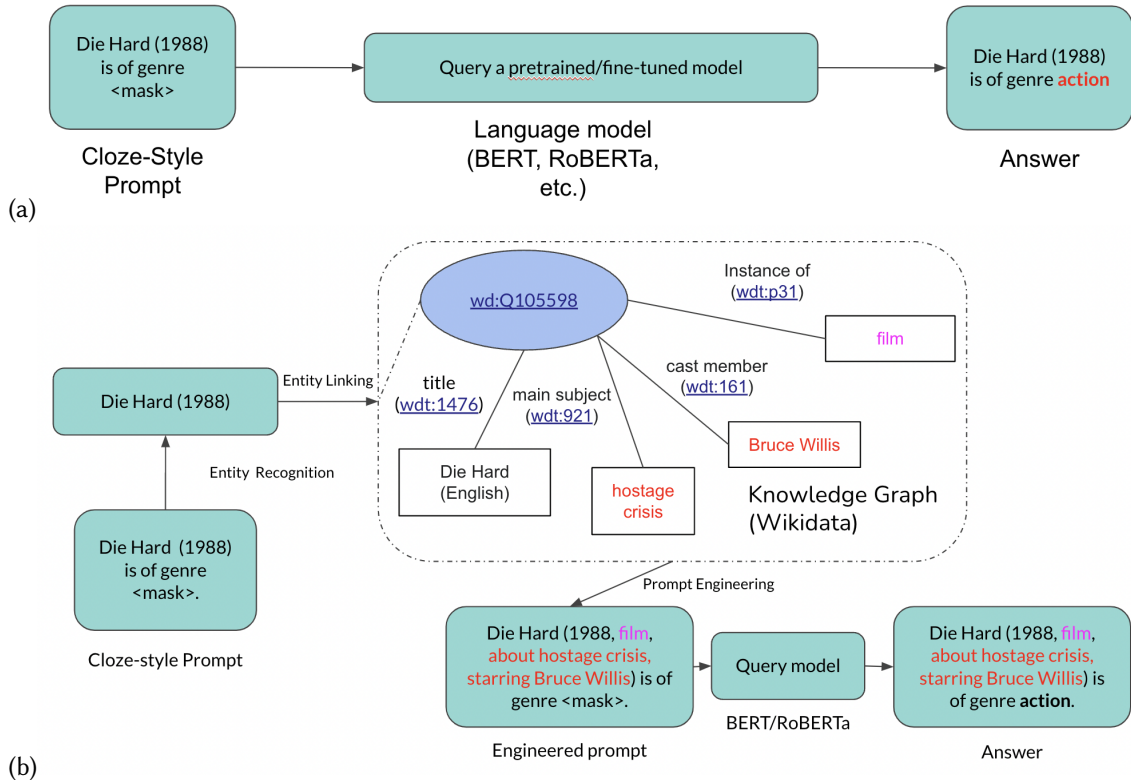


Figure 1: Proposed framework (a) Typical querying setup for a Masked Language Model prediction. (b) Proposed approach to enrich the query using external language.

4. Evaluation

4.1. Dataset

In order to test our approach, we use the BERT [1] and RoBERTa large [20] pre-trained models. The test dataset we are using is a subset of ML25M from IMDB [21]. ML25M contains the title and ground truth genre classification of a range of 54,758 movies. A subset of this dataset was then assembled, as those movies for which the Wikidata properties as listed in Table 1 were present in full. This resulted in a test set of 9,596 movie titles. The Wikidata properties, and thus the corresponding data subset, were selected as a compromise between a large dataset, and a diverse set of domain-relevant Wikidata properties, following exploratory analysis of the ML25M dataset.

4.2. Results

Tables 2 and 3 list the Recall@n scores for each of the prompts described in Section 3.1, for the BERT and RoBERTa large models respectively. For a given model and prompt, Recall@1 and Recall@5 values for each movie are calculated as the fraction of movie ground-truth genres predicted in the highest ranked n PLM mask predictions. The aggregated Recall@n values reported in Tables 2 and 3 are the arithmetic mean Recall@n scores across all movies in the test dataset, with respect to the model and prompt referenced.

One-tailed binomial significance tests have been performed to test whether the higher Recall@n values of the optimum enriched prompts are significantly greater (for $\alpha = 0.05$) than the unenriched prompt, for the corresponding model. The resultant p-values are given in Tables 4 and 5.

prompt	Recall@n scores			
	verbalisation strategy A		verbalisation strategy B	
	R@1	R@5	R@1	R@5
non-enriched	0.136	0.448	0.136	0.448
1	0.139	0.476	0.153	0.487
2	0.161	0.515	0.161	0.498
3	0.092	0.423	0.065	0.305
4	0.024	0.226	0.036	0.258
5	0.017	0.176	0.011	0.090
6	0.004	0.104	0.020	0.062
7	0.055	0.320	0.021	0.214
8	0.047	0.250	0.006	0.065
9	0.020	0.140	0.014	0.083

Table 2

Recall@n scores for the Bert and the movie data subset, averaged over all movies. Verbalisation strategy A and B prompts consist of comma separated and 'and' separated WikiData information, respectively. The greatest Recall@n scores highlighted in bold.

5. Discussion

The results and analysis of Section 4.2 demonstrate that, when considered en-masse, enrichment of prompts with domain relevant information from Wikidata improves cloze-style genre prediction in the movie domain. This is the case for both of the verbalisation strategies trialled.

It is noteworthy, however, that BERT and RoBERTa models behave very differently in terms of both their non-enriched performance, and their performance when subject to varying levels of enrichment. This is demonstrative of the potential for PLM model improvement via prompt enrichment as being highly specific to the model in question. I.e., BERT demonstrates optimum recall performance in aggregate for those enriched prompts with relatively low levels of information enrichment, followed by very rapid reduction in recall@n for further enriched prompts.

	Recall@n scores			
	verbalisation strategy A		verbalisation strategy B	
prompt	R@1	R@5	R@1	R@5
non-enriched	0.065	0.198	0.065	0.198
1	0.096	0.264	0.076	0.204
2	0.114	0.297	0.016	0.068
3	0.057	0.180	0.005	0.031
4	0.010	0.100	0.004	0.038
5	0.034	0.115	0.012	0.043
6	0.013	0.053	0.012	0.044
7	0.191	0.556	0.203	0.534
8	0.163	0.466	0.142	0.389
9	0.184	0.536	0.210	0.576

Table 3

Recall@n scores for the RoBERTa large and the movie data subset, averaged over all movies. Verbalisation strategy A and B prompts consist of comma separated and 'and' separated WikiData information, respectively, as described in Section 3.1 . The greatest Recall@n scores highlighted in bold.

	One-tailed binomial tests p-values			
	verbalisation strategy A		verbalisation strategy B	
	Prompt 2, R@1	Prompt 2, R@5	Prompt 2, R@1	Prompt 2, R@5
non-enriched prompt	(1270, 2084, $7.11e^{-24}$)	(1900, 2698, $6.37e^{-103}$)	(1287, 2121, $2.64e^{-23}$)	(1773, 2817, $1.38e^{-43}$)

Note: * denotes that the p-value is so small that scipy.stats.binom_test function reported the pvalue as 0

Table 4

(x, n, p-values) for separate one-tailed binomial (pairwise) tests which compare the by-movie instances where recall@n differs between optimum enriched prompt predictions and the unenriched prompt prediction for each verbalisation strategy applied to BERT

	One-tailed binomial tests p-values			
	verbalisation strategy A		verbalisation strategy B	
	Prompt 7, R@1	Prompt 7, R@5	Prompt 9, R@1	Prompt 9, R@5
non-enriched prompt	(2534, 2760, 0*)	(5297, 5451, 0*)	(2750, 2937, 0*)	(5627, 5755, 0*)

Note: * denotes that the p-value is so small that scipy.stats.binom_test function reported the pvalue as 0

Table 5

(x, n, p-values) for separate one-tailed binomial (pairwise) tests which compare the by-movie instances where recall@n differs between optimum enriched prompt predictions and the unenriched prompt prediction for each verbalisation strategy applied to RoBERTa

Whereas RoBERTa large demonstrates fluctuating performance relative to the non-enriched prompt, with the greatest performance shown in the more information rich prompts.

It is beyond the scope of this paper to disentangle the role of information variety and the specific information types themselves, as to the influence on prediction outcomes. However, there are preliminary indications of complex interactions. For example as shown in Table 3, prompt 7 (verbalisation strategy A) applied to RoBERTa large shows a huge spike in improved performance over the worst performing prompt 6, which adds the *release date* information. Analysis of a verbalisation strategy A prompt enriched only by *release date* alone, explains

a large portion of the improvement (Recall@1=0.167, Recall@5= 0.48). However, the overall context provided by prompt 7 results in the best performance overall, the one-tailed binomial tests showing prompt 7 significantly more performance for recall@1 (p-value = $4.18e-31$) and recall@5 (p-value = $5.42e-6$). Accordingly, the results are suggestive of further investigate work being required to better understand the interactive effect of information enrichment on whatever model, domain, and task such enriched prompts may be applied to.

6. Conclusion

Given that PLMs are limited in performance for domain-specific cloze-style question answering prompts, in this paper we examine how adding additional context to naive prompts from knowledge graphs can improve the performance of PLMs on a movie genre prediction task. Through our experiments, we show a statistically significant improvement in recall on prompts enriched with information from the Wikidata knowledge graph in comparison to non-enriched prompts on the BERT and RoBERTa large PLMs. As future work, we plan to expand our study to include more domains such as books, music etc. to better understand domain-specific optimum characteristics for enrichment, and cover the same domains as similar previous work [13]. Additionally, we plan to investigate the relationship between different knowledge graph topologies, predicate semantics, queries and graph walks, and the performance of prompts engineered with them at solving various tasks through PLMs.

References

- [1] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, *ArXiv abs/1810.04805* (2019).
- [2] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners, in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), *Advances in Neural Information Processing Systems*, volume 33, Curran Associates, Inc., 2020, pp. 1877–1901. URL: <https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfc4967418bfb8ac142f64a-Paper.pdf>.
- [3] L. Asprino, W. Beek, P. Ciancarini, F. v. Harmelen, V. Presutti, Observing lod using equivalent set graphs: it is mostly flat and sparsely linked, in: *International Semantic Web Conference*, Springer, 2019, pp. 57–74.
- [4] P. Zhang, T. Chai, Y. Xu, Adaptive prompt learning-based few-shot sentiment analysis, *ArXiv abs/2205.07220* (2022).
- [5] T. Kasahara, D. Kawahara, N. Tung, S. Li, K. Shinzato, T. Sato, Building a personalized dialogue system with prompt-tuning, *ArXiv abs/2206.05399* (2022).
- [6] K. Qi, H. Wan, J. Du, H. Chen, Enhancing cross-lingual natural language inference by prompt-learning from cross-lingual templates, in: *Proceedings of the 60th Annual Meeting*

- of the Association for Computational Linguistics (Volume 1: Long Papers), 2022, pp. 1910–1923.
- [7] S. Hu, N. Ding, H. Wang, Z. Liu, J. Wang, J. Li, W. Wu, M. Sun, Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2022, pp. 2225–2240.
 - [8] B. Heinzerling, K. Inui, Language models as knowledge bases: On entity representations, storage capacity, and paraphrased queries, ArXiv abs/2008.09036 (2021).
 - [9] S. Ruder, M. E. Peters, S. Swayamdipta, T. Wolf, Transfer learning in natural language processing, in: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: Tutorials, 2019, pp. 15–18.
 - [10] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, G. Neubig, Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing, arXiv preprint arXiv:2107.13586 (2021).
 - [11] T. Mickus, D. Paperno, M. Constant, K. van Deemter, What do you mean, bert? assessing bert as a distributional semantics model, ArXiv abs/1911.05758 (2019).
 - [12] Z. Jiang, F. F. Xu, J. Araki, G. Neubig, How can we know what language models know?, 2019. URL: <https://arxiv.org/abs/1911.12543>. doi:10.48550/ARXIV.1911.12543.
 - [13] G. Penha, C. Hauff, What does bert know about books, movies and music? probing bert for conversational recommendation, Fourteenth ACM Conference on Recommender Systems (2020).
 - [14] T. Shin, Y. Razeghi, R. L. L. IV, E. Wallace, S. Singh, Eliciting knowledge from language models using automatically generated prompts, ArXiv abs/2010.15980 (2020).
 - [15] S. Min, M. Lewis, H. Hajishirzi, L. Zettlemoyer, Noisy channel language model prompting for few-shot text classification, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2022, pp. 5316–5330.
 - [16] T. Schick, H. Schütze, 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, 2021, pp. 255–269.
 - [17] Y. Lu, M. Bartolo, A. Moore, S. Riedel, P. Stenetorp, Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity, arXiv preprint arXiv:2104.08786 (2021).
 - [18] P. West, C. Bhagavatula, J. Hessel, J. D. Hwang, L. Jiang, R. L. Bras, X. Lu, S. Welleck, Y. Choi, Symbolic knowledge distillation: from general language models to commonsense models, ArXiv abs/2110.07178 (2021).
 - [19] M. Sap, R. Le Bras, E. Allaway, C. Bhagavatula, N. Lourie, H. Rashkin, B. Roof, N. A. Smith, Y. Choi, Atomic: An atlas of machine commonsense for if-then reasoning, in: Proceedings of the AAAI conference on artificial intelligence, volume 33, 2019, pp. 3027–3035.
 - [20] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized BERT pretraining approach, CoRR abs/1907.11692 (2019). URL: <http://arxiv.org/abs/1907.11692>. arXiv:1907.11692.
 - [21] F. M. Harper, J. A. Konstan, The movielens datasets: History and context, *Acm transactions on interactive intelligent systems (tiis)* 5 (2015) 1–19.