Machine Intelligence Research
www.mi-research.net

21(2), April 2024, 217-238
DOI: 10.1007/s11633-023-1416-x

The Life Cycle of Knowledge in Big Language Models: A Survey

Boxi Cao[1,3] Hongyu Lin[1] Xianpei Han[1,2] Le Sun[1,2]

[1] Chinese Information Processing Laboratory, Beijing 100190, China

[2] State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing 100190, China

[3] University of Chinese Academy of Sciences, Beijing 101408, China

Abstract: Knowledge plays a critical role in artificial intelligence. Recently, the extensive success of pre-trained language models (PLMs) has raised significant attention about how knowledge can be acquired, maintained, updated and used by language models. Despite the enormous amount of related studies, there is still a lack of a unified view of how knowledge circulates within language models throughout the learning, tuning, and application processes, which may prevent us from further understanding the connections between current progress or realizing existing limitations. In this survey, we revisit PLMs as knowledge-based systems by dividing the life circle of knowledge in PLMs into five critical periods, and investigating how knowledge circulates when it is built, maintained and used. To this end, we systematically review existing studies of each period of the knowledge life cycle, summarize the main challenges and current limitations, and discuss future directions[1].

Keywords: Pre-trained language model, knowledge acquisition, knowledge representation, knowledge probing, knowledge editing, knowledge application.

Citation: B. Cao, H. Lin, X. Han, L. Sun. The life cycle of knowledge in big language models: a survey. *Machine Intelligence Research*, vol.21, no.2, pp.217–238, 2024. http://doi.org/10.1007/s11633-023-1416-x

1 Introduction

Fundamentally, AI is the science of knowledge – how to represent knowledge and how to obtain and use knowledge.

Nilson (1974)[1]

Knowledge is the key to high-level intelligence. How a model obtains, stores, understands and applies knowledge has long been a critical research topic in machine intelligence. Recent years have witnessed the rapid development of pre-trained language models (PLMs). Through self-supervised pre-training on large-scale unlabeled corpora, PLMs show strong generalization and transferring abilities across different tasks/datasets/settings over previous methods, and therefore have achieved remarkable success in natural language processing[2–7].

The success of pre-trained language models has raised great attention about the nature of their entailed knowledge. There have been numerous studies focusing on how knowledge can be acquired, maintained, and used by pre-trained language models. Along these lines, many novel research directions have been explored. For example, knowledge infusing devotes to injecting explicit structured knowledge into PLMs[8–10]. Knowledge probing aims to evaluate the type and amount of knowledge stored in PLMs' parameters[11–13]. And knowledge editing is dedicated to modifying the incorrect or undesirable knowledge acquired by PLMs[14–16].

Despite the large amount of related studies, current studies primarily focus on one specific stage of knowledge process in PLMs, thereby lacking a unified perspective on how knowledge circulates throughout the entire model learning, tuning, and application phases. The absence of such comprehensive studies makes it hard to better understand the connections between different knowledge-based tasks, discover the correlations between different periods during the knowledge life circle in PLMs, exploit the missing links and tasks for investigating knowledge in PLMs, or explore the shortcomings and limitations of existing studies. For example, while numerous studies attempt to assess the knowledge in language models that are already pre-trained, there are few studies dedicated to investigating why PLMs can learn from pure text without any supervision about knowledge, as well as how PLMs represent or store these knowledge. Meanwhile, many re-

Review
Special Issue on Commonsense Knowledge and Reasoning: Representation, Acquisition and Applications
Manuscript received on October 31, 2022; accepted on January 13, 2023; published online on January 12, 2024
Recommended by Associate Editor Ji-Rong Wen
Colored figures are available in the online version at https://link.springer.com/journal/11633

[1] We openly released a corresponding paper list which will be regularly updated on https://github.com/c-box/KnowledgeLifecycle.

searchers have tried to explicitly inject various kinds of structural knowledge into PLMs, but few studies propose to help PLMs better acquire specific kinds of knowledge from pure text by exploiting the knowledge acquisition mechanisms behind. As a result, related research may be overly focused on several directions but fail to comprehensively understand, maintain and control knowledge in PLMs, and therefore limits the improvements and further application.

In this survey, we propose to systematically review the knowledge-related studies in pre-trained language models from a knowledge engineering perspective. Inspired by research in cognitive science[17, 18] and knowledge engineering[19, 20], we regard pre-trained language models as knowledge-based systems, and investigate the life cycle of how knowledge circulates when it is acquired, maintained and used in pre-trained models[19, 20]. Specifically, we divide the life cycle of knowledge in pre-trained language models into the following five critical periods as shown in Fig. 1:

• **Knowledge acquisition**, which focuses on the procedure of language models learning various knowledge from text or other knowledge sources.

• **Knowledge representation**, which focuses on the underlying mechanism of how different kinds of knowledge are transformed, encoded, and distributed in PLMs' parameters.

• **Knowledge probing**, which aims to evaluate how well current PLMs entailing different types of knowledge.

• **Knowledge editing**, which tries to edit or delete knowledge containing in language models.

• **Knowledge application**, which tries to distill or leverage knowledge in pre-trained language models for practical application.

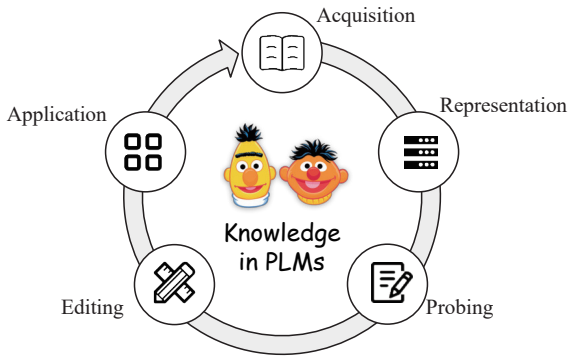

Fig. 1 Five critical periods in life circle of knowledge in language models

For each of these periods, we sort out the existing studies, summarize the main challenges and limitations, and discuss future directions. Based on the unified perspective, we are able to understand and utilize the close connections between different periods instead of considering them as independent tasks. For instance, understanding the knowledge representation mechanism of PLMs is valuable for researchers to design better knowledge acquisition objectives and knowledge editing strategies. Proposing reliable knowledge probing methods could help us find the suitable applications for PLMs, and gain insight into their limitations, thereby facilitating improvement. Through this survey, we are willing to comprehensively conclude the progress, challenges and limitations of current studies, help researchers better understand the whole field from a novel perspective, and shed light on the future directions about how to better regulate, represent and apply the knowledge in language models from a unified perspective.

We summarize our contributions as follows:

1) We propose to revisit pre-trained language models as knowledge-based systems, and divide the life cycle of knowledge in PLMs into five critical periods.

2) For each period, we review existing studies, summarize the main challenges and shortcomings for each direction.

3) Based on this review, we discuss about the limitations of the current research, and shed light to potential future directions.

2 Overview

In this section, we present the overall structure of this survey, describe our taxonomy shown in Fig. 2 in detail, and discuss the topics in each critical period.

Knowledge acquisition is the knowledge learning procedure of language models. Currently, there are two main sources for knowledge acquisition: the plain text data and the structured data. For acquiring knowledge from text data, LMs typically conduct self-supervised learning on large-scale text corpora[2–4, 6]. This survey will focus on the methods and mechanisms of how pre-trained language models obtaining knowledge from pure texts[21–23]. For acquiring knowledge from structured data, current researches focus on knowledge injection from different kinds of structured data into PLMs. The primary categories of structured data contains entity knowledge[8, 24, 25], factual knowledge[9, 26–28], commonsense knowledge[29–31, 32] and linguistic knowledge[33–36]. We will discuss all of them in Section 3.

Knowledge representation aims to investigate how language models encode, store and represent knowledge in their dense parameters. The investigation about the knowledge representation mechanisms will aid in a better understanding and control of knowledge in PLMs, and may also inspire researchers for better understanding the knowledge representation in human brains. Currently, the strategies for knowledge representation analysis in PLMs include gradient-based[37, 38], causal inspired[39], attention-based[40, 41, 12], and layer-wise[12, 42, 43] methods. We will discuss them in Section 4.

Knowledge probing aims to evaluate how well current PLMs entailing specific types of knowledge. Currently, two primary strategies are used to probe the

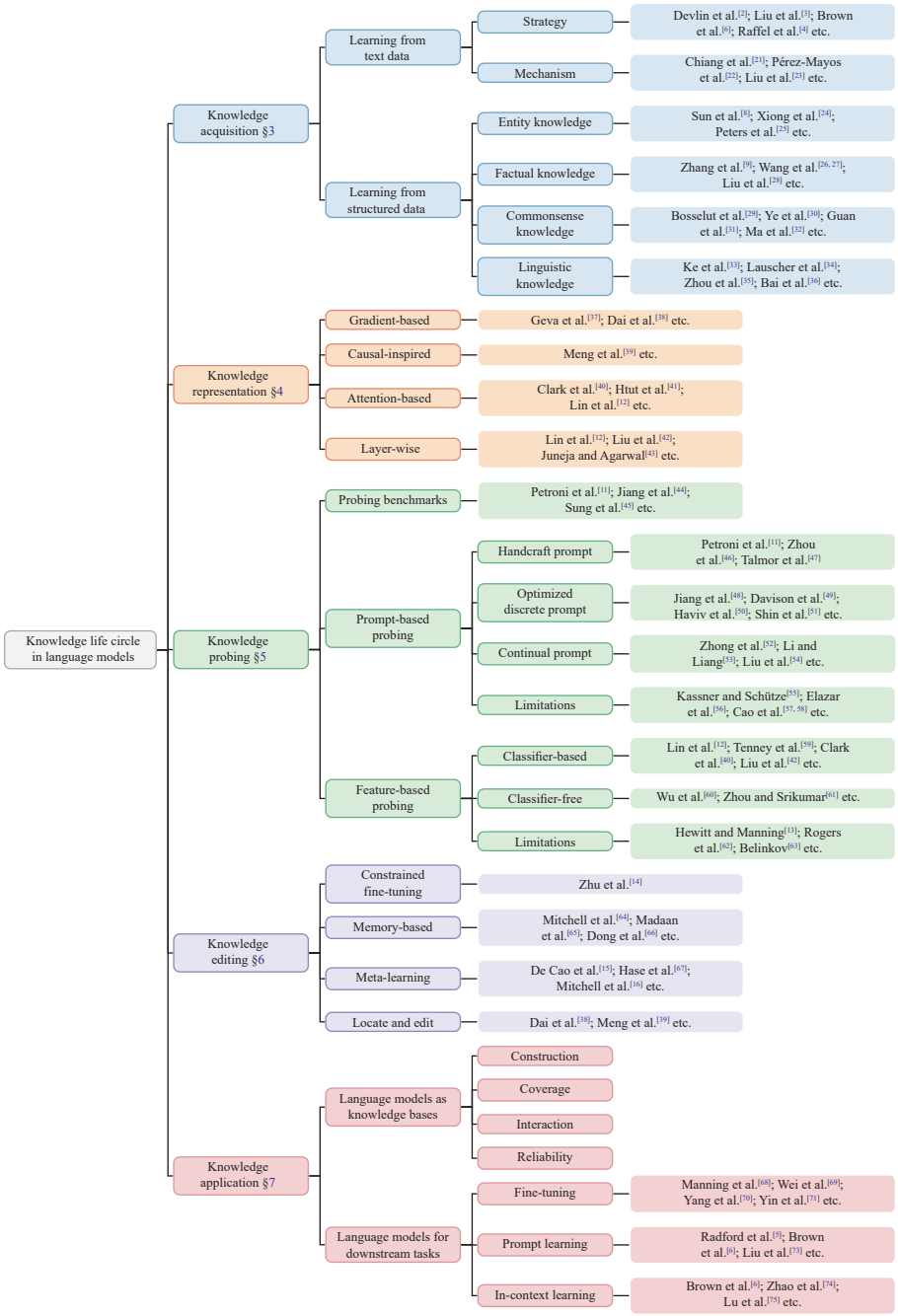

Fig. 2 Typology of knowledge life circle in big language models

knowledge in PLMs: 1) Prompt-based probing, which usually constructs knowledge-instructed prompt, then query PLMs using these natural language expressions[11, 44–46, 76]. For example, querying PLMs with "The capital of France is __." to evaluate whether PLMs have stored the corresponding knowledge ⟨France, capital, Paris⟩. Meanwhile, to improve PLMs' performance, a series of studies devote to optimizing prompts in both discrete[48–51] and continual space[52–54]. Despite the widely application of prompt-based probing, lots of studies also point out that there still exist some pending issues such as inconsistent[55, 56, 58, 77], inaccurate[52, 57, 78] and unreliable[57, 79], and question the quantity results of prompt-based probing. 2) Feature-based probing, which normally freezes the parameters of original PLMs, and evaluates PLMs on probing tasks based on their internal representation or attention weights. We categorize existing feature-based probing studies into classifier-based probing[12, 40, 42, 59] and classifier-free probing[60, 61] according to whether an additional classifier is introduced. Since most methods introduce additional parameters or training data, the main shortcoming of feature-based probing is whether the results should attribute to knowledge in PLMs or probing task learned by additional probes. We

will discuss them in Section 5.

Knowledge editing aims to modify the incorrect knowledge or delete the undesirable information in PLMs. Because of inevitable mistakes learned by PLMs and the update of knowledge, reliable and effective knowledge editing approaches are essential for the sustainable application of PLMs. Current approaches include constrained fine-tuning[14], memory-based[64–66], meta learning inspired[15, 16, 67] and location-based methods[38, 39]. We will discuss them in Section 6.

Knowledge application aims to distill or leverage specific knowledge from PLMs to benefit further applications. Currently, there are two main kinds of application paradigms for knowledge in PLMs: 1) Language models as knowledge bases (LMs-as-KBs), which regards language models as dense knowledge bases that can be directly queried with natural language to obtain specific types of knowledge[11, 48, 57, 80–83]. And we provide a comprehensive comparison between structured knowledge bases and LMs-as-KBs[82] from four aspects, including construction, coverage, interaction and reliability; 2) Language models for downstream task, which directly uses PLMs entailing specific kinds of knowledge in downstream NLP tasks via fine-tuning[68–71], prompt-learning[5, 6, 73] and in-context learning[6, 74, 75]. We will discuss them in Section 7.

3 Knowledge acquisition

During the knowledge acquisition period, pre-trained language models learn knowledge from different knowledge sources. In this section, we categorize and describe knowledge acquisition strategies according to knowledge sources, and then discuss the future directions.

3.1 Learning from text data

Currently, pre-trained language models usually acquire various knowledge from pure text through self-supervised learning on a large-scale text corpus. In this section, we will first introduce several widely used learning objectives[84, 85], and then discuss the learning mechanisms behind them.

Causal language modeling aims to autoregressively predict the next token in the input sequence, which is the most popular pre-training tasks[5, 6, 86, 87] and has demonstrated excellent effectiveness in capturing context dependency and text generation paradigms. One limitation of causal language modeling is unidirectional, which can only capture contextual information from left to right.

Masked language modeling aims to mask some tokens in the input randomly, and then predict the masked token conditioned on the rest of sequence[2, 3]. Unlike causal language modeling, which can only obtain information in a unidirectional manner, masked language modeling can capture contextual information from both left-to-right and right-to-left directions.

Seq2seq masked language modeling uses an encoder-decoder architecture for pre-training, which first feeds the encoder with masked sequence, and the decoder is supposed to predict the masked tokens autoregressively[4, 88].

Denoising autoencoder first corrupts the input sequence with randomly mask symbols, then feed the input into a bidirectional encoder, and the likelihood of the whole original input is calculated with an auto-regressive decoder[7].

Although PLMs are pre-trained without any supervision from external knowledge sources, they have been shown to capture a diverse range of knowledge within their parameters, such as linguistic knowledge[12, 13, 41, 42, 59, 90, 91], semantic knowledge[59, 92, 93] and world knowledge[49, 76, 94–98].

Intuitively, PLMs learn such knowledge because they can abstract, generalize and store the implicit knowledge in the text through self-supervised learning. Unfortunately, the underlying mechanism of how and why PLMs acquire or forget knowledge still remains to be explored. And it will be valuable to understand the behaviors of PLMs and inspire better knowledge acquisition strategies.

To understand the underlying mechanisms, some studies dive into the dynamics of LMs′ pre-training procedure. Many researchers study the training dynamics of neural networks. For example, Achille et al.[99] try to figure out whether there exist critical periods in the learning process of neural networks. Liu et al.[23] devote to finding a mathematical solution for the semantic development in deep linear networks. Other studies[100, 101] analyze the training dynamics of LSTM[102] with techniques such as SVCCA[103].

While most existing studies focus on neural networks with relatively simple architectures, only a few studies consider knowledge in large-scale pre-trained language models. Chiang et al.[21] first systematically investigate the knowledge acquisition process during the training of ALBERT[104]. Specifically, they study the syntactic knowledge, semantic knowledge, and world knowledge development during pre-training, and find that the learning process varies across knowledge, and having more pre-trained steps could not necessarily increase the knowledge in PLMs. Pérez-Mayos et al.[22] investigate the effect of the size of the pre-trained corpus on the syntactic ability of the RoBERTa[3] model, and find that models pre-trained on more data typically contain more syntactic knowledge and perform better in related downstream tasks. Liu et al.[23] also investigate the knowledge acquisition process of RoBERTa[3] on various knowledge. And find that compared with linguistic knowledge which can be learned quickly and robustly, world knowledge is learned slowly and domain-sensitively.

3.2 Learning from structured data

Apart from acquiring knowledge from pure text,

PLMs can also acquire knowledge by injecting explicit structured knowledge into them. In this section, we review these studies according to the category of structured knowledge sources.

Entity knowledge. To learn entity knowledge explicitly, lots of studies propose entity-guided tasks for language model pre-training. For example, Sun et al.[8] and Shen et al.[105] use entity-level masking to enhance language models, which first recognize named entities in a sentence, and then all the sub-words within an entity are masked and predicted at once. Xiong et al.[24] present replaced entity detection, which randomly replaces the named entities in a sentence with another mention of the same entity or other entities of the same type, and LMs are supposed to determine which entities are replaced. Yamada et al.[106] treat words and entities as independent tokens, and conduct mask language modeling separately to learn both contextualized word representation and entity representation. Févry et al.[107] combine the mention detection and entity linking pre-training objectives with mask language modeling to match the entities in text with specific entity memories. In addition to the entity mentions themselves, researchers have also introduced other meta-information such as entity description to further assist the entity knowledge learning[108, 109]. Another efficient way to enrich PLMs′ text representation with entity knowledge is utilizing word-to-entity attention[25, 106].

Factual knowledge. In structured knowledge bases, factual knowledge is generally represented as triples (subject entity, relation, object entity). For a long time, researchers have been dedicated to aiding PLMs to acquire more factual knowledge to perform better on downstream tasks. On the one hand, introducing knowledge graph embedding into the pre-training procedure could be effective. Zhang et al.[9] propose an aggregator to combine the corresponding knowledge embedding of the entities in text and token embedding. Wang et al.[26] co-train mask language modeling and knowledge graph embedding objectives, which could produce both informative text and knowledge embedding. On the other hand, some studies propose designing factual knowledge-guided auxiliary tasks. Wang et al.[27] add an adapter to infuse knowledge into PLMs without updating the original parameters. The adapter is trained with predication prediction to determine the relation type between tokens. Qin et al.[110] propose the entity discrimination tasks to predict the object entity given subject entity and relation, as well as relation discrimination tasks to predict the semantic connection between relation pairs. Banerjee and Baral[111] directly pre-train language model on the knowledge graph, the model is given two elements of a knowledge triple to predict the rest one. Liu et al.[28] argue that incorporating a whole knowledge base into PLMs might induce the knowledge noise issue, and propose to learn from a specific sub-graph related to each input sentence. Moreover, Soares et al.[112] propose to learn relational knowledge solely from entity-link text through "matching in the blank" objective, which first replaces the entities in text with blank symbols and then brings the relation representations closer when they have the same pair of entities.

Commonsense knowledge. One of the most common strategies for PLMs learning commonsense knowledge is converting the knowledge to natural language expressions before learning. Bosselut et al.[29], Guan et al.[31], Shwartz et al.[113] first transfer the commonsense knowledge triples to natural language with prompt, then pre-train LMs on these knowledge-augmented data. Ye et al.[30] post-train LMs on commonsense QA datasets created by AMS (align, mask, select). Ma et al.[32] transform structured commonsense knowledge into natural language questions for model learning.

Linguistic knowledge. By designing the corresponding pre-training tasks, PLMs could also learn linguistic knowledge explicitly, such as sentiment knowledge[33, 114], lexical knowledge[34, 35, 115], syntax knowledge[10, 35, 36], etc. For example, to equip LMs with sentiment knowledge, Ke et al.[33] first label each word with a POS tag and sentiment polarity, and then incorporate both the word-level and sentence-level sentiment label with the mask language modeling objective. Similarly, Tian et al.[114] first mine sentiment knowledge from unlabeled data based on pointwise mutual information (PMI), and then conduct pre-training tasks such as sentiment masking, sentiment word prediction and word polarity prediction with these sentiment information. As for lexical knowledge, Lauscher et al.[34] first acquire word similarity information from WordNet[116] and BabelNet[117], and then add word relation classification tasks in addition to BERT′s original pre-training tasks. Levine et al.[115] also introduce the lexical information from WordNet and add a masked-word prediction task. To incorporate dependency knowledge with PLMs, Song et al.[118] construct a dependency matrix for attention alignment calibration and a fusion module to integrate dependency information. Explicitly learning syntax knowledge also raises the researchers′ attention, Sachan et al.[10] investigate infusing syntax knowledge by either adding a syntax-GNN on the output of transformers or incorporating with text embedding using attention. To further capture the syntax knowledge, Bai et al.[36] use multiple attention networks, with each one encoding one relation from the syntax tree.

3.3 Discussions and future work

As we mentioned above, there have been extensive studies for better knowledge acquisition of language models, and most of them focus on infusing existing structured knowledge sources into PLMs. The learning from text data methods can be easily scaled, and the knowledge sources are easily obtained. But the underlying

mechanism is still mostly unclear, the knowledge acquisition process is implicitly and thus is hard to control, and may lead to inconsistent prediction, undesirable bias and unforeseen risks. The learning from structured data methods can explicitly inject knowledge into PLMs, but are limited by the cost, domain, scale and quality of knowledge sources. Furthermore, since the knowledge injection methods are often specialized to specific kinds of knowledge, it is often difficult to extend or produce new knowledge.

Furthermore, because all knowledge in PLMs are implicitly encoded as parameters, it is often very difficult to control and validate the knowledge acquisition process. There are also several studies such as retrieval-based PLMs, focusing on retrieving related knowledge or context to enhance original PLMs[119−121], rather than injecting knowledge into PLMs′ parameters.

Several future directions of knowledge acquisition in PLMs may lie in: 1) For the knowledge acquisition from existing structured knowledge sources, it is critical to develop universal knowledge injection methods which can uniformly injecting different types of knowledge from different knowledge sources, and ensure continuous learning and avoid catastrophic forgetting in the meantime. 2) For the knowledge acquisition from pure text data, it is helpful to fully understand the underlying mechanism of knowledge learning in PLMs, and develop effective knowledge learning algorithms which can learn specific knowledge from text data in a controllable and predicable way. 3) Furthermore, it is also important to build comprehensive benchmarks for investigating and assessing the knowledge acquisition process of PLMs.

4 Knowledge representation

Knowledge representation studies investigate how pretrained language models encode, transform and store the acquired knowledge. In PLMs, knowledge is encoded to dense vector representations and held in their distributed parameters, but how each kind of knowledge is encoded, transformed, and stored into the parameters is still unclear and needs further investigation. Currently, a few studies have investigated the knowledge representation in language models, and we will first review these studies according to their analysis techniques.

4.1 Analyzing knowledge representations in PLMs

Currently, the analyzing approaches for knowledge representation in PLMs can be classified into four categories: gradient-based, causal-inspired, attention-based and layer-wise methods. The first three methods aim to locate specific knowledge in PLMs′ corresponding neurons or attention heads, and the layer-wise methods hypothesize that knowledge is represented in different layers of PLMs.

Gradient-based. Dai et al.[38] first introduce the concept of knowledge neurons, which are neurons in transformer[122] related to certain factual knowledge. Specifically, they hypothesize the knowledge neurons are located in feed-forward networks, which are considered as key-value memories[37]. Then by feeding the LM with knowledge-expressing prompts such as “Michael Jordan was born in [MASK]”, the corresponding knowledge neuron is identified as the neurons in the feed-forward networks with higher attribution scores, which are calculated based on integrated gradients.

Causal-inspired. Meng et al.[39] identify knowledge neurons as the neuron activations in transformers that have the strongest causal effect on predicting certain factual knowledge. Such neurons are located through a causal mediation analysis. Specifically, they calculate the causal effect on factual prediction by comparing probability variation of object prediction between the clean and corrupted token embedding. Their experiments also demonstrate that the mid-layer feed-forward modules play a decisive role in factual knowledge representation.

Attention-based. In addition to the feed-forward layers, the attention heads are also be considered as representations which may encode the knowledge-related information. Clark et al.[40], Htut et al.[41] investigate the linguistic knowledge encoded in attention heads, and find that while some individual attention heads are associated with specific aspects of syntax, the linguistic knowledge is distributed and represented by multiple attention heads. Lin et al.[12] find that PLMs′ attention weights could encode syntactic properties such as subject-verb agreement and reflexive dependencies, and higher layers represent these syntactic properties more accurately.

Layer-wise. Lin et al.[12] conduct a layer-wise probing for linguistic knowledge, which trains a specific classifier for each layer, and find that the lower layers encode the positional information of tokens, and higher layers encode more compositional information. Liu et al.[42] analyze the layerwise transferability of PLMs on a wide range of tasks and find that the middle layers usually have better performance and transferability. Wallat et al.[123] propose to probe the captured factual knowledge with LAMA[11] of each layer in PLMs, and find that a significant amount of knowledge is stored in the intermediate layers. Juneja and Agarwal[43] also conduct a layer-wised factual knowledge analysis based on knowledge neuron[38], and demonstrate that most relational knowledge (e.g., Paris is the capital of “some nation”.) can be attributed to the middle layers, which would be refined into facts (e.g., Paris is the capital of France.) in the last few layers.

4.2 Discussions and future works

The above studies reach some consensus about know-

ledge representation in PLMs, including: 1) Factual knowledge can be associated with feedforward modules in middle or higher layers. 2) Linguistic knowledge is distributed and represented in multiple attention heads, while a single attention head can only associate with a specific aspect of linguistics. 3) The lower layers of PLMs often encode the coarse-grained and general information of knowledge, while the fine-grained and task-specific knowledge are mostly stored in higher layers. These findings are valuable for us to understand knowledge representation in language models but are also limited to specific knowledge types or model architectures. Therefore, the knowledge representation in PLMs is still an open problem which needs further exploration.

In the future, several directions of knowledge representation in PLMs may lie in the following: 1) Because knowledge representation is a long-standing concern in cognitive science, neuroscience, psychology, and artificial intelligence, it is helpful to borrow ideas from other related areas and design cognitively-inspired analysis methods. 2) Current knowledge representation studies in PLMs mostly focus on a specific type of knowledge and often result in local and specific conclusions. It is important to comprehensively investigate different types of knowledge together, e.g., compare the differences and commonalities of knowledge representations of different knowledge types, pretraining tasks, or model architectures, and come up with more universal and insightful conclusions.

5 Knowledge probing

Knowledge probing aims to assess how well pre-trained language models entail different kinds of knowledge. A comprehensive and accurate assessment of PLMs′ knowledge can help us identify and understand language models′ capabilities and deficiencies, allow a fair comparison between LMs with different architectures and pre-training tasks, guide the improvement of a specific model, and select suitable models for different real-world scenarios. In this section, we will first introduce existing benchmarks for knowledge probing, then introduce the representative prompt-based and feature-based probing methods and analyze their corresponding limitations, and discuss future directions.

5.1 Benchmarks for knowledge probing

To assess the knowledge in PLMs, lots of benchmarks have been proposed to probe various knowledge contained in PLMs, for example, linguistic knowledge[12, 59, 91, 93, 124], syntactic knowledge[13, 40], factual knowledge[11, 44, 45, 125], commonsense knowledge[46, 76], etc. Table 1 summarizes several representative knowledge probing benchmarks.

5.2 Prompt-based knowledge probing

Prompt-based probing is one of the most popular approaches for knowledge probing. To evaluate whether LMs know a specific knowledge such as the birthplace of Michael Jordan, we could query LMs with knowledge queries such as “Michael Jordan was born in __.”, where “was born in” is a prompt for a specific type of knowledge. As shown in Table 1, prompt-based probing has

Table 1 Summary about some representative knowledge probing benchmarks

Method	Benchmarks	Knowledge type	Formulation
Prompt-based	LM diagnostics[93]	Linguistic	Text filling
	BLiMP[124]	Linguistic	Sentence scores comparison
	LAMA[11]	Factual, commonsense	Text filling
	X-FACTR[44]	Factual, multilingual	
	Multilingual LAMA[125]	Factual, multilingual	
	Bio LAMA[45]	Factual, biological	
	CAT[46]	Commonsense	Sentence scores comparison text filling
	NumerSense[97]	Commonsense, numerical	
	oLMPICS[47]	Reasoning	Multiple choices
Feature-based	Open sesame[12]	Linguistic	Diagnostic classifier and attention
	LKT[42]	Linguistic	Token or token pair labeling
	NPI probe[91]	Linguistic	Probing classifier
	Edge probe[59]	Linguistic, semantic	Edge probing
	MDL probe[127]	Linguistic	Minimum description length
	Structural probe[13]	Syntactic	Structural probing
	Physical commonsense[76]	Commonsense, physical	Probing classifier

been widely used in benchmarks such as LAMA[11], oLMpics[47], LM diagnostics[93], BIG-bench[128], etc.

For prompt-based probing, the main challenge is how to design effective prompts which are suitable for different kinds of knowledge and different PLMs. In the following, we will introduce the typical prompt types for knowledge probing and discuss their limitations.

5.2.1 Prompt development

Handcraft prompt. Early methods often manually write prompts for different kinds of knowledge. There are two primary advantages of manually created prompts: the readability without the need of any other resources or training. For example, LAMA[11] manually creates one cloze-style prompt for each relation, which is used to probe the factual knowledge in language models. CAT[46] reframes the instances in existing commonsense datasets into paired sentences with task-specific prompts, and determines whether PLMs contain specific commonsense knowledge by comparing the sentence scores, e.g., "money can be used to buy cars" VS. "money can be used to buy stars". oLMpics[47] converts the probing tasks for reasoning ability into multi-choice questions with manually created prompts, and compare the LMs' probability of candidate choices.

Optimized discrete prompt. Despite the mentioned advantages, Jiang et al.[48] argue that handcraft prompts could be sub-optimal. Therefore, a series of studies have been proposed to optimize the prompts in a discrete space so that PLMs could achieve better performance. Jiang et al.[48] propose a mining-based method in order to find prompts with higher performance from text corpus. They first retrieve potential prompts which contain both the subject and object entity, then select prompts using a validation dataset. Davison et al.[49] select prompt from a handcrafted candidate set according to the log-likelihood calculated by LMs. Haviv et al.[50] propose a paraphrasing-based method, where each query is first reframed by a trained rewriter and then fed into PLMs. Shin et al.[51] propose an automatic prompt generation method based on gradient-guided search, where a prompt is iteratively updated from "[MASK]" token by maximizing the label likelihood of training instances.

Continual prompt. Although the prompts generated by Shin et al.[51] are discrete text, they are very difficult to be understood by humans. Therefore, several studies directly search better-performed prompts on continual space rather than confining to discrete space, i.e., representing prompts as dense vectors. Continual prompts have shown good performance for knowledge probing, and further extensions include handcraft prompts initialization[52], adding continual prompts on both input and transformer blocks[53] or adding LSTM layers above the input embeddings[54].

5.2.2 Limitations of prompt-based probing

Although prompts have been widely used to probe the knowledge in PLMs, there are still lots of pending issues unresolved, which make the probing results unstable and the assessment of knowledge in PLMs unreliable.

Inconsistent. Prompt-based probing have been shown often result in inconsistent results due to prompt selection, instance verbalization, negation, etc. Firstly, Elazar et al.[56] find semantically equivalent prompts may result in different predictions, Cao et al.[58] further find that PLMs would prefer specific prompts with the same linguistic regularity with the pre-training corpus, such a prompt preference will significantly affect the probing results, and result in inconsistent comparisons between PLMs. Besides prompts, the instance verbalization process also leads to inconsistent predictions. For example, when we ask BERT "The capital of the U.S. is [MASK]", the answer is Washington, but when we replace the U.S. with its alias America, the prediction will change to Chicago. In addition, PLMs also exhibit inconsistency when facing negation[55, 77]. For instance, PLMs would generate highly similar predictions between a fact ("Birds can fly") and its incorrect negation ("Birds cannot fly")[55]. Jang et al.[77] conduct the negation experiments on PLMs of varying sizes and various downstream tasks, and find that not only PLMs cannot well understand negation prompts, but also show an inverse scaling law.

Inaccurate. The performance of PLMs under prompt-based probing may also be overestimated. Poerner et al.[78] find that many samples in the probing datasets could be easily "guessed" by only relying on the surface form association. For example, the object entity is a substring of the subject entity (e.g., "Apple Watch is produced by Apple"). Furthermore, the training dataset for prompt optimization may correlate with probing dataset, which results in spurious correlations[52] and the performance improvements may come from these spurious correlations. Cao et al.[57] also find that many prompts with better performance are prompts which over-fit to answer distributions, rather than a better semantic description of the target relation.

Unreliable. To reach a faithful probing result, it is essential to understand why PLMs make a specific prediction. However, studies find that PLMs do not always make predictions based on specific knowledge. In that case, the knowledge probing results could be unreliable. Cao et al.[57] find that the prompts but not the answers dominate the prediction distribution of PLMs, resulting in severely prompt-biased probing conclusions. Li et al.[79] conduct a causal-inspired analysis and find that PLMs' predictions rely more on words that are close in position and frequently co-occur, rather than those related to knowledge.

Bias analysis. While lots of studies conduct empirical experiments on the biases in prompt-based probing, few have investigated the source and interpretation of these biases. Several studies employ causal analysis for bias analysis, which has been widely used to identify undesirable biases and fairness concerns[129–133]. Cao et al.[58]

propose a causal analysis framework to identify, interpret and eliminate biases that exist in prompt-based probing with a theoretical guarantee. Similarly, Elazar et al.[134] also propose a causal framework to estimate the causal effects of the data statistics in training corpus on the factual predictions of PLMs. Finlayson et al.[135] apply causal mediation analysis to investigate the syntactic agreement mechanisms in PLMs.

5.3 Feature-based knowledge probing

Feature-based knowledge probing is also widely used to probe the knowledge in PLMs, where the parameters of original PLMs are frozen, and the probing tasks are accomplished based on the internal representation or attention weights produced by PLMs. In this section, we introduce and discuss the feature-based probing approaches.

5.3.1 Classifier-based probing

Classifier-based probing trains a classifier to predict specific knowledge properties on the top of the fixed PLMs, and assesses the effectiveness of PLMs using the classifier's performance[63]. Such approaches are first proposed to evaluate the linguistic properties (e.g., morphological, syntactic) associated with static embeddings[136, 137], and have been widely used to probe the linguistic knowledge[12, 59, 40, 42, 13] and semantic knowledge[59, 92, 138, 42] in PLMs. Popular classifiers include linear classifier, logistics regression, multi-layer perceptron, etc.

5.3.2 Classifier-free probing

Since the results and conclusions of classifier-based methods are dependent on the training quality and selection of the classifier, some studies have developed feature-based probing approaches without an additional classifier. For example, Wu et al.[60] propose perturbed masking, which calculates an impact matrix through a two-stage perturbation, where the matrix captures the impacts of a token on the prediction of another token, and is further used for the syntactic probe. Zhou and Srikumar[61] introduce DirectProbe, which directly probes the geometric properties of PLMs' representation without an additional classifier. Clark et al.[40] probe syntactic knowledge in language models by investigating the attention weights without a classifier, e.g., analyzing the most attended word of the given token.

5.3.3 Limitations of feature-based probing

There are two main limitations of current feature-based probing approaches[62, 63]. The first limitation concerns the attribution of results, which is originally pointed out by Hewitt and Manning[13]. While most probes introduce additional training data and parameters, it's difficult to attribute evaluation results to the knowledge in PLMs, or the probe itself, which may learn to perform the probing task. The second limitation pertains to the inconsistency between different probe designs for the same type of knowledge. There are various probe selections for each kind of knowledge, but the probe results between simple probes like linear classifier or complex probes could be inconsistent.

5.4 Discussions and future works

With the growing scale and abilities of big language models, the comprehensive, accurate and reliable measurements of the actual knowledge and capabilities of LMs become increasingly important. However, the accurate, robust and reliable probing approach is still an open problem. Firstly, as we discussed above, both prompt-based probing and feature-based probing have their own limitations, which might result in unreliable or even contradicting conclusions. Secondly, most existing benchmarks are specialized to specific knowledge types and specific model architectures.

In the future, the main directions of knowledge probing may lie in: 1) Comprehensive benchmark construction. As we demonstrate in Table 1, current knowledge probing benchmarks are mostly too specialized, which may lead to inconsistent, biased or unreliable results. Therefore, it is critical to build a comprehensive and unbiased benchmark. 2) Debiased probing approaches. Currently, prompt-based probing is the dominant knowledge probing methods due to its simplicity. However, there still exist lots of issues in prompt-based probing. Therefore, the design of unbiased datasets and better probing frameworks is another useful direction worth investigating.

6 Knowledge editing

Knowledge editing is the process which modifies the stored knowledge in pre-trained language models, either by replacing it with new knowledge (e.g., changing the current prime minister of the UK to Rishi Sunak) or by removing it entirely (e.g., some personal privacy information). There are two primary motivations for editing knowledge in language models: 1) Even the state-of-the-art language models (e.g., ChatGPT[2]) could learn lots of incorrect knowledge; 2) Many facts are time-sensitive, requiring regular updates to their corresponding knowledge.

Unfortunately, editing knowledge in PLMs poses significant challenges. Firstly, naive solutions such as retraining are often impractical due to the massive size of large-scale language models. Secondly, due to the black box and non-linear nature of PLMs, any minor modification might result in a significant undesirable change in model predictions. As a result, it can be challenging to precisely edit the target knowledge.

To promote the development of relevant studies, De Cao et al.[15] formulate three desiderata for knowledge editing methods: 1) **Generality**: The method is able to edit the language models already pre-trained without the need for specialized re-training. 2) **Reliability**: The method is supposed to successfully edit knowledge re-

[2] https://openai.com/blog/chatgpt/

quired modification while not influencing the rest of knowledge in LMs. 3) **Consistency**: The modification should be consistent across paraphrases with equivalent semantics (e.g., Michael Jordan was born in [MASK]. VS. The birthplace of Michael Jordan is [MASK].) and relevant knowledge required modification accordingly (e.g., Rishi Sunak becomes the prime minister of the UK. VS. Liz Truss is not the prime minister of the UK.).

In this section, we divide current strategies for knowledge editing into four categories and the summary of comparisons between these approaches is shown in Table 2. In the following, we will describe and discuss these methods.

6.1 Constrained fine-tuning

The naive solution to edit knowledge in a PLM is to re-train it using the updated training dataset, but such a naive solution is computationally expensive and may be impractical because PLMs are involved. Therefore, a better solution is to fine-tune PLMs only on a small subset which only contains the target samples. However, such a method may suffer from catastrophic forgetting, and affects the rest knowledge which is not intended to be edited. Therefore, Zhu et al.[14] propose to modify the knowledge in PLMs with constrained fine-tuning, specifically, they use an $\mathcal{L}_2$ or $\mathcal{L}_\infty$ normalization to constrain the parameters change of models. Furthermore, they find that only fine-tuning the initial and final layers while keeping the rest of the model frozen yields better performance than finetuning the whole model. However, in deep neural networks like PLMs, even a minor change of the parameters could change the model′s predictions on a lot of samples. Therefore, such methods could potentially affect other knowledge stored in PLMs which is not required modification.

6.2 Memory-based editing

Instead of directly modifying parameters of PLMs, another natural solution is to maintain a knowledge cache which stores all new knowledge, and replace the original predictions when an input hits the cache. However, a symbolic knowledge cache may suffer from robustness issues, i.e., the inputs with the same meaning can differ in natural language expressions, therefore they may result in different predictions. To address this problem, Mitchell et al.[64] propose a memory-based approach for knowledge editing. Specifically, the model contains five modules: an edit memory that stores the modified knowledge, a classifier, a counterfactual model, and the frozen original language model. Given an input, the classifier determines whether it hits a sample in the edit memory, and the counterfactual model′s prediction will overrule the original language model′s prediction if it hits a memory cache. This method is effective but does not actually edit the knowledge encoded in the parameters of language models, thus cannot benefit downstream tasks. Meanwhile, Dong et al.[66] add additional trainable parameters in the feed-forward module of PLMs, which are trained on a modified knowledge dataset while the original parameters are frozen. They also demonstrate that the modified knowledge could benefit related QA tasks. Moreover, Madaan et al.[65] introduce the users′ feedback for PLMs′ error correction. Specifically, they maintain a memory of models′ mistake and users′ feedback, which enhance the model to produce updated prompt and avoid similar mistakes.

Table 2 Comparisons between existing knowledge editing approaches. "Online edit" refers to quickly editing an individual target knowledge. "Batch edit" refers to editing a set of target knowledge simultaneously. "Downstream benefit" refers to the potential for the modified knowledge to be utilized by the edited language model for downstream tasks. "Unforeseen side effects" refers to the impact of knowledge editing on the language model beyond the modification of target knowledge.

Approach	Knowledge support	Training required	Online edit	Batch edit	Downstream benefit	Unforeseen side effects
Constrained tuning						
FTM[14]	Factual	YES	NO	YES	Potential	YES
Memory-based						
SERAC[64]	Factual, QA	YES	YES	YES	NO	NO
MEM-PROMPT[65]	Linguistic, ethics	NO	YES	YES	Potential	Unlikely
CALINET[66]	Factual	YES	NO	YES	Potential	YES
Meta-learning						
KNOWEDITOR[15]	Factual	YES	YES	YES	Potential	YES
MEMD[16]	Factual	YES	YES	YES	Potential	YES
Locate and edit						
Knowledge neuron[38]	Factual	NO	YES	NO	NO	YES
ROME[39]	Factual	NO	YES	NO	Potential	Possible

6.3 Meta-learning-based editing

Sinitsin et al.[140] first propose editable training to conduct model editing based on meta-learning, which aims to train the model parameters to suit model editing. By constraining the training objective, the editing procedure could be accomplished under k gradient step while ensuring reliability, locality, and efficiency. However, such a method is not practical for pre-trained language models since it requires expensive specialized retraining. A different strategy is to utilize a hyper network, which uses one network to generate the weights of another network[141]. De Cao et al.[15], Hase et al.[67] train a hyper-network to predict the parameter changes for each data point, with the constraint of editing target knowledge without affecting others. Although computationally efficient, Mitchell et al.[16] argue that this method fails to edit very large models, and propose model editor networks with gradient decomposition (MEND). Specifically, by decomposing the gradient of standard fine-tuning into a low-rank form, they could train multiple MLPs to generate local model parameter changes, without damaging models′ predictions on unrelated knowledge. Experiments show that MEND can be applied to large pre-trained models for fast model editing. One limitation of existing meta-learning-based methods is that their robustness and generalization are still questionable, as they ensure locality by constraining the parameter space change or the predictions on specific datasets. In that case, the knowledge that requires no modifications or the knowledge that is related to edited knowledge but not paraphrasing could also be incorrect.

6.4 Locate and edit

Based on the assumption that "knowledge is locally stored in PLMs", the "locate and edit" strategy first locates the parameters corresponding to specific knowledge, and edit them by directly replacing with updated ones. This approach is also introduced in Section 4.1. Dai et al.[38] present a case study of factual knowledge editing in PLMs with corresponding knowledge neurons. By directly modifying the value of knowledge neurons, they achieve knowledge editing with a relatively low but non-trivial success rate. Although the editing procedure is straightforward once the corresponding knowledge neuron is located, this method has not proved its effectiveness on large-scale editing or the effects of unrelated knowledge. Similarly, Meng et al.[39] first connect the knowledge required modification with a key-value pair in one of the middle MLP layers, and modify the corresponding knowledge by directly updating the key-value pair. Since these methods are based on the locality hypothesis of factual knowledge, which has not been widely confirmed yet, the changes in certain parameters may affect irrelevant knowledge and lead to unexpected results.

6.5 Discussions and future works

To utilize pre-trained language models as a sustainable knowledge resource, the precise, effective, reliable and consistent knowledge editing is essential. However, as discussed above, all current editing methods have their own limitations. Therefore, it is worthwhile to enhance current methods and develop new knowledge editing strategies.

In the future, several useful directions of knowledge editing may lie in: 1) **Broader range of target knowledge.** As shown in Table 2, current studies mostly focus on the editing of factual knowledge, which is relatively easy to formalize and evaluate. In the future, researchers could explore the editing methods towards other kinds of knowledge, and develop universal approaches which can edit all kinds of knowledge in the same way. 2) **Comprehensive evaluation.** Currently, most knowledge editing studies are evaluated using metrics such as editing success rate on target knowledge, predictions invariance rate on unrelated knowledge for assessing generality, and accuracy on paraphrases of target knowledge for assessing consistency. However, we find that these metrics are limited to comprehensively evaluate the knowledge editing capability of different approaches. For instance, most evaluations only sample unrelated knowledge from the same distribution of target knowledge. However, the influence of a knowledge edit could be much broader, e.g., affecting the performance on downstream tasks or the knowledge from other distributions and categories. In addition, as mentioned in Mitchell et al.[16], most studies measure the consistency of samples generated through back translation, which ignores the knowledge affected by knowledge editing except the paraphrases, e.g., the country with the largest population would be affected by the population modification of the countries. Therefore, it is important to design comprehensive benchmark which can better assess the capabilities of editing strategies. 3) **More effective editing approaches.** Ideally, a knowledge editing approach should satisfy the desiderata of generality, reliability and consistency, and can handle large-scale and individual knowledge editing tasks with high efficiency. To this end, we may borrow ideas from other fields, such as meta-learning, continual learning, and life-long learning. Furthermore, it is useful to connect knowledge editing studies with knowledge representation studies (Section 4).

7 Knowledge application

Knowledge application studies how to effectively distill and leverage the knowledge in PLMs for other applications. Specifically, we divide knowledge applications into two categories: language models as knowledge bases and language models for downstream tasks, and in following we describe them in detail.

7.1 Language models as knowledge bases

The impressive performance of large-scale pretrained language models, as well as the potentially enormous amount of implicitly stored knowledge, raises extensive attention about using language models as an alternative to conventional structured knowledge bases (LMs-as-KBs)[11, 48, 57, 80–83].

Unfortunately, along with the promising advantages and potentials compared with structured knowledge bases, there also exist intrinsic flaws for language models as knowledge base[82], which are summarized in Table 3. In following we describe them in detail.

Table 3 The comparisons between conventional structured knowledge bases and using language models as knowledge bases (LMs-as-KBs). Part of this table is inspired by Razniewski et al.[82] The advantages are marked in bold. From the table, we can easily find that although LMs-as-KBs are more advantageous on construction and coverage, the critical current limitations of interaction and reliability significantly hinder its real-word applications, and far from substitution of structured knowledge bases.

Perspectives	Structured KB	LMs-as-KBs
Construction		
Ontology/schema	Pre-defined	**Open-ended** ☺
Process	Pipline	**End-to-end** ☺
Human effort	Data annotation	**Self-supervised** ☺
Expert knowledge	Common	**Not required** ☺
Coverage		
Domain	Constrained	**Open** ☺
Amount	Limited	Potential
Knowledge fusing	Complex	**Easy** ☺
Interaction		
Query	Structured	**Natural language** ☺
Prediction	**Deterministic** ☺	Probabilistic
Rejection	**Yes** ☺	Hard
Editing	**Easy** ☺	Limited
Reliability		
Ambiguity	**Low** ☺	High
Correctness	**Relatively high** ☺	Questionable
Current practicality	**Extensive** ☺	Limited yet

Construction procedure is one of the biggest advantages of LMs-as-KBs compared with structured KBs. Constructing large-scale structured KBs such as Freebase[142] and Wikidata[143] often requires extremely complex pipelines[11], e.g., ontology construction, knowledge acquisition, knowledge verification, knowledge fusion, knowledge storage, and knowledge population. Such a complex pipeline involves lots of NLP techniques, including ontology engineering, entity linking, entity recognition, relation extraction, entity matching and so on. And each technique requires corresponding expert knowledge, supervised data and human efforts. Moreover, due to the pipeline nature, error propagation is always a critical issue.

In contrast, the knowledge of language models can be easily learned from pure text using self-supervised learning, without any explicit supervision signal (Section 3.1). Furthermore, the construction procedure is end-to-end, therefore no ontology engineering, expert knowledge, or human annotations are needed.

Coverage is another big advantage of LMs-as-KBs. Traditional structured KBs are often limited by its predefined schemas, and the difficulty of acquiring knowledge further limits their coverage. In comparison, by directly representing knowledge in parameters, there is no schema limitations for LMs-as-KBs. And all knowledge is learned from un-annotated text corpus, therefore the knowledge coverage is mostly only determined by the coverage of pre-training corpus.

The above advantages make LMs-as-KBs an extremely attractive and promising idea. However, there are also some intrinsic flaws which hinder LMs from fully substituting structured KBs.

Interaction with structured KB and LMs-as-KBs are quite different. Structured KBs often use structural querying methods such as SPARQL[144], e.g., querying the birthplace of Michael Jordan using ⟨Michael Jordan, Birthplace,?⟩. In the case of language model-based KBs, the queries are mostly natural language expressions such as "The birthplace of Michael Jordan is [MASK]".

Compared with structural queries, natural language-based queries are more natural and friendly for users. However, structured KBs can return deterministic answers (e.g., Brooklyn), but LM-based KBs can only generate candidates with different probabilities (e.g., ⟨Brooklyn, 0.8⟩). The probabilistic predictions may be incorrect, inconsistent and confusing. Furthermore, structured KBs can identify the queries they cannot answer, but current LM-based KBs can hardly reject the queries it cannot answer, thus resulting in the knowledge hallucination problem. Concretely, if we query some knowledge that is not stored in a structured KB, the answer could be blank when no tuples are matched. However, no matter what we ask, language models will always "guess" the answers, even such knowledge is never learned by LMs. Although there are some naive solutions to this problem such as rejecting answers with a low probability, this is still an open problem currently.

Finally, it is difficult to edit knowledge in LM-based KBs, as discussed in Section 6. In comparison, it is easy to add, modify and delete knowledge in structured KBs.

Reliability is another concern for LMs-as-KBs. The first problem is ambiguity. In structured KBs, all entities and facts have their own IDs (e.g., Q89 for Apple the fruit and Q312 for Apple Inc. in Wikidata), therefore there is no ambiguity problem. However, in LM-based KBs, all pieces of knowledge are represented as natural language expressions and will therefore suffer from the ambiguity problem of natural language. For example, do "U.S.A" and "America" represent the same entity in a language model? Previous studies have observed that such verbalization requirements will result in prompt preference bias and instance verbalization bias in LMs-as-KBs[58]. The consistency of predictions is another drawback of LMs-as-KBs, i.e., a LM-based KB may return different answers to the semantically equivalent queries.

7.2 Language models for downstream tasks

Besides using language models as knowledge bases, the knowledge in PLMs can also benefit many downstream tasks in different ways. Fig. 3 shows three main paradigms and we describe them in detail.

7.2.1 Fine-tuning

Fine-tuning is a common way to leverage knowledge in language models, which learns to distill and leverage knowledge by further tuning PLMs using task-specific datasets. Firstly, implicitly learned knowledge from text has been recognized as one of the main reasons for PLMs' remarkable performance and strong generalization ability across so many NLP tasks[68–71]. Secondly, many studies have shown that injecting knowledge into language models can lead to better performance on downstream tasks. For instance, the integration of entity knowledge into pre-trained language models (PLMs) has shown potential for improving the performance of various language understanding tasks (Sun et al.[8]; Shen et al.[105]); Similarly, incorporating factual knowledge into PLMs has been found to enhance their performance in tasks such as relation extraction and entity typing, etc.[9, 26–28]; Furthermore, the incorporation of linguistic knowledge with PLMs has demonstrated performance improvements on benchmarks such as GLUE[10, 36, 115].

7.2.2 Prompt learning

Prompt-based learning is another way to leverage the knowledge in PLMs for downstream tasks. For example, to classify the sentiment polarity of the sentence "Best movie ever.", we can add a prompt and transform the input into "Best movie ever. It is __.". And the polarity can be determined by comparing the PLMs' prediction probability between candidate answers "good" and "bad". By selecting appropriate prompts, PLMs have been shown competitive zero-shot performance on some downstream tasks without any supervised training[5, 6, 73].

Because handcraft prompts often suffer from unstable performance across different prompts and cannot utilize the information from supervised data, many prompt optimization approaches have been proposed to acquire better-performing prompts[73], such as paraphrasing[48, 50], gradient-based search[51], model generation[145], knowledge enhanced[146], etc. Furthermore, prompt-tuning, which adds some trainable vectors to the inputs as continuous prompts, while keeping the parameters of LMs freezing, has achieved competitive performance with fine-tuning[53, 54, 148, 149]. In addition to optimizing single prompts, ensembling[48, 150], compositing[151], or decoupling[152] multiple prompts could also improve model performance. Moreover, prompt has also been applied to data augmentation[153], domain adaptation[154], debiasing[155] and so on.

More recently, instruction-tuning, which pretrains LMs on a wide range of datasets given the natural language description of tasks as instructions, has achieved significant performance and generalization ability improvements of language models[86, 156–158].

7.2.3 In-context learning

Applications. Currently, the parameters of PLMs have been scaled to 175 B (e.g., GPT-3[6], OPT[159], BLOOM[3]) or even larger (e.g., PaLM[160]), making the computational expense of fine-tuning and prompt-tuning infeasible for most researchers. Therefore, tuning-free in-context learning has become one of the most popular approaches to apply the knowledge in large-scale PLMs in downstream tasks[161]. For instance, for the sentiment classification task, in-context learning will first sample several demonstrations, such as (what a horrible meal, negative), and combine them with the original query. In this way, the input becomes "What a horrible meal. It is bad. [SEP] Best movie ever. It is __." The provided demonstrations offer extra information about the task and enable PLMs to utilize the analogy ability to predict the correct answer. Incontext learning has achieved good performance on lots of downstream tasks such as language understanding[6, 74, 162–164], data generation[165–167], or reasoning[168–170].

Bias problem. One drawback of in-context learning is the bias problem, i.e., the performance is sensitive to demonstration selections, demonstration orders, label distribution of demonstrations and prompt selection, etc.[74, 75, 171]. Therefore, to achieve better performance of in-context learning, Zhao et al.[74] first propose to estimate the biases by feeding the model with an uninformative input (e.g., [MASK] or N/A), and then calibrate the prediction probabilities uniformly distributed for eliminating the models' bias towards specific answers. For demonstration selection, Gao et al.[145], Liu et al.[171] propose to select demonstrations that are semantically close to the input query. Rubin et al.[172] train a dense retriever on LM-scored datasets to select demonstrations. Su et al.[173] in-

[3] https://huggingface.co/bigscience/bloom

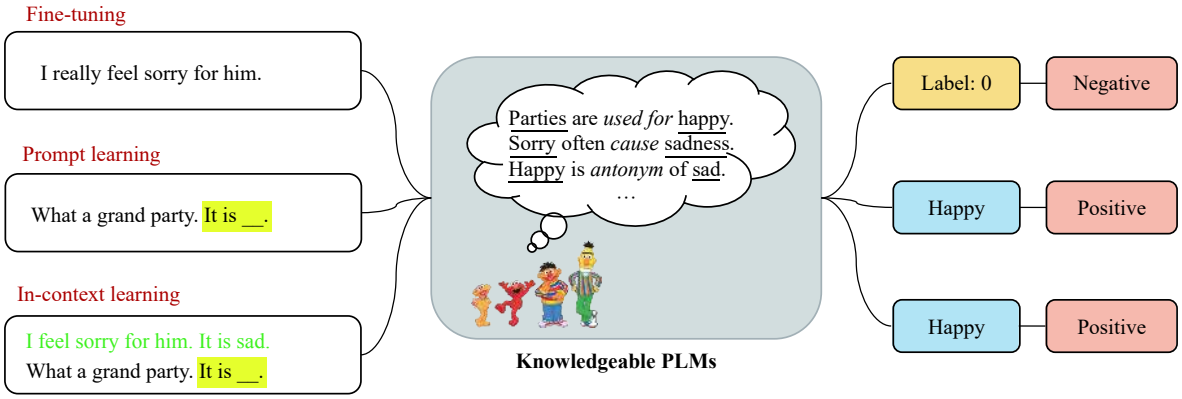

Fig. 3 The primary paradigms that apply the knowledge in PLMs to downstream tasks

troduce a graph-based selection method to ensure the demonstration's diversity and representativeness. For demonstration sort, Lu et al.[75] first construct a development dataset by sampling from language models, and then use entropy-based metrics to determine the optimal demonstration permutation. For prompt selection, Gao et al.[145] use a language model to generate candidate prompts and select ones with better performance on the development set.

Mechanism. Although in-context learning has been widely applied on various downstream tasks, its underlying mechanism is still unclear. Reynolds and McDonell[72] find that zero-shot prompting sometimes can significantly outperform in-context learning, and argue that the additional demonstrations do not help PLMs to learn a new task, but rather locate the task they have already learned. Cao et al.[57] investigate the in-context learning for knowledge probing, and find that the demonstrations can only provide type-level guidance but not factual information. Min et al.[89] find that randomly replacing the demonstrations' labels hardly affects the performance, and show that the effectiveness of in-context learning relies more on the label space and input distribution restriction provided by demonstrations rather than the precise input label mapping. Chan et al.[126] find that only when the data includes both burstiness and large-scale of rarely occurring classes, in-context learning capability can emerge in transformer model. von Oswald et al.[139] investigate the connections between in-context learning and gradient descent, and demonstrate the similarity between in-context learning and the gradient-based few-shot learning.

7.3 Discussions and future works

Leveraging knowledge in PLMs is both promising and challenging. On the one hand, it is obvious that the large amount of implicit knowledge stored in PLMs will benefit different downstream tasks. On the other hand, all current application paradigms have their own limitations. For instance, the consistency and reliability of LMs-as-KBs hinder PLMs to replace structured KBs. Moreover, fine-tuning, prompt learning and in-context learning methods often suffer from catastrophic forgetting, computational cost, inconsistent and unstable predictions, social bias, etc.

To address these challenges, several main future directions of knowledge application may lie in the following: 1) For LMs-as-KBs, we need to propose specific pre-training approaches to address current shortcomings in consistency and reliability. 2) For LMs for downstream tasks, we suggest explore more application strategies, such as new tuning-free methods to address the computational cost issue and black-box tuning[147] methods to tune pre-trained language models without access to their parameters.

8 Conclusions

In this survey, we conduct a comprehensive review about the life circle of knowledge in pre-trained language models, including knowledge acquisition, knowledge representation, knowledge probing, knowledge editing and knowledge application. We systematically review related studies for each period, discuss the advantages and limitations of different methods, summarize the main challenge, and present some future directions. We believe this survey will benefit researchers in many areas such as language models, knowledge graph, knowledge base, etc.

Acknowledgements

This research work is supported by the National Natural Science Foundation of China (No. 62 122 077) and CAS Project for Young Scientists in Basic Research, China (No. YSBR-040).

Declarations of conflict of interest

The authors declared that they have no conflicts of interest to this work.

References

[1] N. J. Nilsson. Artificial intelligence. In *Proceedings of the*

6th IFIP Congress 1974, Stockholm, Sweden, pp. 778–801, 1974.

[2] J. Devlin, M. W. Chang, K. Lee, K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics, Minneapolis, USA, pp. 4171–4186, 2019. DOI: 10.18653/v1/N19-1423.

[3] Y. H. Liu, M. Ott, N. Goyal, J. F. Du, M. Joshi, D. Q. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach, [Online], Available: https://arxiv.org/abs/1907.11692, 2019.

[4] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Q. Zhou, W. Li, P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, vol. 21, no. 1, Article number 140, 2020.

[5] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, vol. 1, no. 8, Article number 9, 2019.

[6] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. 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. M. 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 *Proceedings of the 34th International Conference on Neural Information Processing Systems*, Vancouver, Canada, Article number 159, 2020.

[7] M. Lewis, Y. H. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7871–7880, 2020. DOI: 10.18653/v1/2020.acl-main.703.

[8] Y. Sun, S. H. Wang, Y. K. Li, S. K. Feng, X. Y. Chen, H. Zhang, X. Tian, D. X. Zhu, H. Tian, H. Wu. ERNIE: Enhanced representation through knowledge integration, [Online], Available: https://arxiv.org/abs/1904.09223, 2019.

[9] Z. Y. Zhang, X. Han, Z. Y. Liu, X. Jiang, M. S. Sun, Q. Liu. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, pp. 1441–1451, 2019. DOI: 10.18653/v1/P19-1139.

[10] D. Sachan, Y. H. Zhang, P. Qi, W. L. Hamilton. Do syntax trees help pre-trained transformers extract information? In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 2647–2661, 2021. DOI: 10.18653/v1/2021.eacl-main.228.

[11] F. Petroni, T. Rocktäschel, S. Riedel, P. Lewis, A. Bakhtin, Y. X. Wu, A. Miller. Language models as knowledge bases? In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, Hong Kong, China, pp. 2463–2473, 2019. DOI: 10.18653/v1/D19-1250.

[12] Y. J. Lin, Y. C. Tan, R. Frank. Open sesame: Getting inside BERT's linguistic knowledge. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, Florence, Italy, pp. 241–253, 2019. DOI: 10.18653/v1/W19-4825.

[13] J. Hewitt, C. D. Manning. A structural probe for finding syntax in word representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, USA, pp. 4129–4138, 2019. DOI: 10.18653/v1/N19-1419.

[14] C. Zhu, A. S. Rawat, M. Zaheer, S. Bhojanapalli, D. L. Li, F. Yu, S. Kumar. Modifying memories in transformer models, [Online], Available: https://arxiv.org/abs/2012.00363, 2020.

[15] N. De Cao, W. Aziz, I. Titov. Editing factual knowledge in language models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Punta Cana, Dominican Republic, pp. 6491–6506, 2021. DOI: 10.18653/v1/2021.emnlp-main.522.

[16] E. Mitchell, C. Lin, A. Bosselut, C. Finn, C. D. Manning. Fast model editing at scale, [Online], Available: https://arxiv.org/abs/2110.11309, 2022.

[17] P. G. Zimbardo, F. L. Ruch. *Psychology and Life*, 9th ed., Scott, Foresman, 1975.

[18] P. S. Churchland, T. J. Sejnowski. Perspectives on cognitive neuroscience. *Science*, vol. 242, no. 4879, pp. 741–745, 1988. DOI: 10.1126/science.3055294.

[19] R. Studer, V. Richard Benjamins, D. Fensel. Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, vol. 25, no. 1–2, pp. 161–197, 1998. DOI: 10.1016/S0169-023X(97)00056-6.

[20] G. Schreiber, H. Akkermans, A. Anjewierden, R. de Hoog, N. R. Shadbolt, W. Van de Velde, B. J. Wielinga. *Knowledge Engineering and Management: The CommonKADS Methodology*, Cambridge, USA: MIT Press, 2000.

[21] C. H. Chiang, S. F. Huang, H. Y. Lee. Pretrained language model embryology: The birth of ALBERT. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 6813–6828, 2020. DOI: 10.18653/v1/2020.emnlp-main.553.

[22] L. Pérez-Mayos, M. Ballesteros, L. Wanner. How much pretraining data do language models need to learn syntax? In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Punta Cana, Dominican Republic, pp. 1571–1582, 2021. DOI: 10.18653/v1/2021.emnlp-main.118.

[23] Z. Y. Liu, Y. Z. Wang, J. Kasai, H. Hajishirzi, N. A. Smith. Probing across time: What does RoBERTa know and when? In *Proceedings of Findings of the Association for Computational Linguistics: EMNLP 2021*, Association for Computational Linguistics, Punta Cana, Dominican Republic, pp. 820–842, 2021. DOI: 10.18653/v1/2021.findings-emnlp.71.

[24] W. H. Xiong, J. F. Du, W. Y. Wang, V. Stoyanov. Pretrained encyclopedia: Weakly supervised knowledge-pretrained language model. In *Proceedings of the 8th International Conference on Learning Representations*, Addis Ababa, Ethiopia, 2020.

[25] M. E. Peters, M. Neumann, R. Logan, R. Schwartz, V. Joshi, S. Singh, N. A. Smith. Knowledge enhanced con-

textual word representations. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, Hong Kong, China, pp. 43–54, 2019. DOI: 10.18653/v1/D19-1005.

[26] X. Z. Wang, T. Y. Gao, Z. C. Zhu, Z. Y. Zhang, Z. Y. Liu, J. Z. Li, J. Tang. KEPLER: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 176–194, 2021. DOI: 10.1162/tacl_a_00360.

[27] R. Z. Wang, D. Y. Tang, N. Duan, Z. Y. Wei, X. J. Huang, J. S. Ji, G. H. Cao, D. X. Jiang, M. Zhou. K-Adapter: Infusing knowledge into pre-trained models with adapters. In *Proceedings of the Findings of the Association for Computational Linguistics*, pp. 1405–1418, 2021. DOI: 10.18653/v1/2021.findings-acl.121.

[28] W. J. Liu, P. Zhou, Z. Zhao, Z. R. Wang, Q. Ju, H. T. Deng, P. Wang. K-BERT: Enabling language representation with knowledge graph. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence, the 32nd Innovative Applications of Artificial Intelligence Conference, the 10th AAAI Symposium on Educational Advances in Artificial Intelligence,* New York, USA, pp. 2901–2908, 2020.

[29] A. Bosselut, H. Rashkin, M. Sap, C. Malaviya, A. Celikyilmaz, Y. Choi. COMET: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, pp. 4762–4779, 2019. DOI: 10.18653/v1/P19-1470.

[30] Z. X. Ye, Q. Chen, W. Wang, Z. H. Ling. Align, mask and select: A simple method for incorporating commonsense knowledge into language representation models, [Online], Available: https://arxiv.org/abs/1908.06725, 2019.

[31] J. Guan, F. Huang, Z. H. Zhao, X. Y. Zhu, M. L. Huang. A knowledge-enhanced pretraining model for commonsense story generation. *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 93–108, 2020. DOI: 10.1162/tacl_a_00302.

[32] K. X. Ma, F. Ilievski, J. Francis, Y. Bisk, E. Nyberg, A. Oltramari. Knowledge-driven data construction for zero-shot evaluation in commonsense question answering. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence, the 33rd Conference on Innovative Applications of Artificial Intelligence, the 11th Symposium on Educational Advances in Artificial Intelligence,* pp. 13507–13515, 2021.

[33] P. Ke, H. Z. Ji, S. Y. Liu, X. Y. Zhu, M. L. Huang. SentiLARE: Sentiment-aware language representation learning with linguistic knowledge. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 6975–6988, 2020. DOI: 10.18653/v1/2020.emnlp-main.567.

[34] A. Lauscher, I. Vulić, E. M. Ponti, A. Korhonen, G. Glavaš. Specializing unsupervised pretraining models for word-level semantic similarity. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 1371–1383, 2020. DOI: 10.18653/v1/2020.coling-main.118.

[35] J. R. Zhou, Z. S. Zhang, H. Zhao, S. L. Zhang. LIMIT-BERT: Linguistic informed multi-task BERT. In *Proceedings of the Findings of the Association for Computational Linguistics*, pp. 4450–4461, 2020. DOI: 10.18653/v1/2020.findings-emnlp.399.

[36] J. G. Bai, Y. J. Wang, Y. R. Chen, Y. M. Yang, J. Bai, J. Yu, Y. H. Tong. Syntax-BERT: Improving pre-trained transformers with syntax trees. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 3011–3020, 2021. DOI: 10.18653/v1/2021.eacl-main.262.

[37] M. Geva, R. Schuster, J. Berant, O. Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Punta Cana, Dominican Republic, pp. 5484–5495, 2021. DOI: 10.18653/v1/2021.emnlp-main.446.

[38] D. M. Dai, L. Dong, Y. R. Hao, Z. F. Sui, B. B. Chang, F. R. Wei. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, Dublin, Ireland, pp. 8493–8502, 2022. DOI: 10.18653/v1/2022.acl-long.581.

[39] K. Meng, D. Bau, A. Andonian, Y. Belinkov. Locating and editing factual associations in GPT, [Online], Available: https://arxiv.org/abs/2202.05262, 2022.

[40] K. Clark, U. Khandelwal, O. Levy, C. D. Manning. What does BERT look at? An analysis of BERT's attention. In *Proceedings of the ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, Florence, Italy, pp. 276–286, 2019. DOI: 10.18653/v1/W19-4828.

[41] P. M. Htut, J. Phang, S. Bordia, S. R. Bowman. Do attention heads in BERT track syntactic dependencies? [Online], Available: https://arxiv.org/abs/1911.12246, 2019.

[42] N. F. Liu, M. Gardner, Y. Belinkov, M. E. Peters, N. A. Smith. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, USA, pp. 1073–1094, 2019. DOI: 10.18653/v1/N19-1112.

[43] J. Juneja, R. Agarwal. Finding patterns in knowledge attribution for transformers, [Online], Available: https://arxiv.org/abs/2205.01366, 2022.

[44] Z. B. Jiang, A. Anastasopoulos, J. Araki, H. B. Ding, G. Neubig. X-FACTR: Multilingual factual knowledge retrieval from pretrained language models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 5943–5959, 2020. DOI: 10.18653/v1/2020.emnlp-main.479.

[45] M. Sung, J. Lee, S. Yi, M. Jeon, S. Kim, J. Kang. Can language models be biomedical knowledge bases? In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Punta Cana, Dominican Republic, pp. 4723–4734, 2021. DOI: 10.18653/v1/2021.emnlp-main.388.

[46] X. H. Zhou, Y. Zhang, L. Y. Cui, D. D. Huang. Evaluating commonsense in pre-trained language models. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence, the 32nd Innovative Applications of Artificial Intelligence Conference, the 10th AAAI Symposium on Educational Advances in Artificial Intelligence,* New York, USA, pp. 9733–9740, 2020.

[47] A. Talmor, Y. Elazar, Y. Goldberg, J. Berant. oLMpics-on what language model pre-training captures. *Transactions of the Association for Computational Linguistics*,

vol. 8, pp. 743–758, 2020. DOI: 10.1162/tacl_a_00342.

[48] Z. B. Jiang, F. F. Xu, J. Araki, G. Neubig. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 423–438, 2020. DOI: 10.1162/tacl_a_00324.

[49] J. Davison, J. Feldman, A. Rush. Commonsense knowledge mining from pretrained models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, Hong Kong, China, pp. 1173–1178, 2019. DOI: 10.18653/v1/D19-1109.

[50] A. Haviv, J. Berant, A. Globerson. BERTese: Learning to speak to BERT. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 3618–3623, 2021. DOI: 10.18653/v1/2021.eacl-main.316.

[51] T. Shin, Y. Razeghi, R. L. Logan IV, E. Wallace, S. Singh. AutoPrompt: Eliciting knowledge from language models with automatically generated prompts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 4222–4235, 2020. DOI: 10.18653/v1/2020.emnlp-main.346.

[52] Z. X. Zhong, D. Friedman, D. Q. Chen. Factual probing is[MASK]: Learning vs. learning to recall. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5017–5033, 2021. DOI: 10.18653/v1/2021.naacl-main.398.

[53] X. L. Li, P. Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pp. 4582–4597, 2021. DOI: 10.18653/v1/2021.acl-long.353.

[54] X. Liu, Y. N. Zheng, Z. X. Du, M. Ding, Y. J. Qian, Z. L. Yang, J. Tang. GPT understands, too, [Online], Available: https://arxiv.org/abs/2103.10385, 2021.

[55] N. Kassner, H. Schütze. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7811–7818, 2020. DOI: 10.18653/v1/2020.acl-main.698.

[56] Y. Elazar, N. Kassner, S. Ravfogel, A. Ravichander, E. Hovy, H. Schütze, Y. Goldberg. Measuring and improving consistency in pretrained language models. *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 1012–1031, 2021. DOI: 10.1162/tacl_a_00410.

[57] B. X. Cao, H. Y. Lin, X. P. Han, L. Sun, L. Y. Yan, M. Liao, T. Xue, J. Xu. 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*, pp. 1860–1874, 2021. DOI: 10.18653/v1/2021.acl-long.146.

[58] B. X. Cao, H. Y. Lin, X. P. Han, F. C. Liu, L. Sun. Can prompt probe pretrained language models? Understanding the invisible risks from a causal view. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, Dublin, Ireland, pp. 5796–5808, 2022. DOI: 10.18653/v1/2022.acl-long.398.

[59] I. Tenney, P. Xia, B. Chen, A. Wang, A. Poliak, R. T. McCoy, N. Kim, B. Van Durme, S. R. Bowman, D. Das, E. Pavlick. What do you learn from context? Probing for sentence structure in contextualized word representations. In *Proceedings of the 7th International Conference on Learning Representations*, New Orleans, USA, 2019.

[60] Z. Y. Wu, Y. Chen, B. Kao, Q. Liu. Perturbed masking: Parameter-free probing for analyzing and interpreting BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4166–4176, 2020. DOI: 10.18653/v1/2020.acl-main.383.

[61] Y. C. Zhou, V. Srikumar. DirectProbe: Studying representations without classifiers. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5070–5083, 2021. DOI: 10.18653/v1/2021.naacl-main.401.

[62] A. Rogers, O. Kovaleva, A. Rumshisky. A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 842–866, 2020. DOI: 10.1162/tacl_a_00349.

[63] Y. Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, vol. 48, no. 1, pp. 207–219, 2022. DOI: 10.1162/coli_a_00422.

[64] E. Mitchell, C. Lin, A. Bosselut, C. D. Manning, C. Finn. Memory-based model editing at scale. In *Proceedings of the 39th International Conference on Machine Learning*, Baltimore, USA, pp. 15817–15831, 2022.

[65] A. Madaan, N. Tandon, P. Clark, Y. M. Yang. Memory-assisted prompt editing to improve GPT-3 after deployment. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Abu Dhabi, UAE, pp. 2833–2861, 2022.

[66] Q. X. Dong, D. M. Dai, Y. F. Song, J. J. Xu, Z. F. Sui, L. Li. Calibrating factual knowledge in pretrained language models. In *Proceedings of the Findings of the Association for Computational Linguistics*, Abu Dhabi, UAE, pp. 5937–5947, 2022.

[67] P. Hase, M. Diab, A. Celikyilmaz, X. Li, Z. Kozareva, V. Stoyanov, M. Bansal, S. Iyer. Do language models have beliefs? Methods for detecting, updating, and visualizing model beliefs, [Online], Available: https://arxiv.org/abs/2111.13654, 2021.

[68] C. D. Manning, K. Clark, J. Hewitt, U. Khandelwal, O. Levy. Emergent linguistic structure in artificial neural networks trained by self-supervision. *Proceedings of the National Academy of Sciences of the United States of America*, vol. 117, no. 48, pp. 30046–30054, 2020. DOI: 10.1073/pnas.1907367117.

[69] X. K. Wei, S. Wang, D. J. Zhang, P. Bhatia, A. Arnold. Knowledge enhanced pretrained language models: A comprehensive survey, [Online], Available: https://arxiv.org/abs/2110.08455, 2021.

[70] J. Yang, G. Xiao, Y. L. Shen, W. Jiang, X. Y. Hu, Y. Zhang, J. H. Peng. A survey of knowledge enhanced pretrained models, [Online], Available: https://arxiv.org/abs/2110.00269, 2021.

[71] D. Yin, L. Dong, H. Cheng, X. D. Liu, K. W. Chang, F. R. Wei, J. F. Gao. A survey of knowledge-intensive NLP with pre-trained language models, [Online], Available: https://arxiv.org/abs/2202.08772, 2022.

[72] L. Reynolds, K. McDonell. Prompt programming for large language models: Beyond the few-shot paradigm. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, ACM, Yokohama, Japan, Article

number 314, 2021. DOI: 10.1145/3411763.3451760.

[73] P. F. Liu, W. Z. Yuan, J. L. Fu, Z. B. Jiang, H. Hayashi, G. Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, vol. 55, no. 9, Article number 195, 2023. DOI: 10.1145/3560815.

[74] Z. H. Zhao, E. Wallace, S. Feng, D. Klein, S. Singh. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 12697–12706, 2021.

[75] Y. Lu, M. Bartolo, A. Moore, S. Riedel, P. Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, Dublin, Ireland, pp. 8086–8098, 2022. DOI: 10.18653/v1/2022.acl-long.556.

[76] M. Forbes, A. Holtzman, Y. Choi. Do neural language representations learn physical commonsense? In *Proceedings of the 41th Annual Meeting of the Cognitive Science Society, CogSci 2019: Creativity + Cognition + Computation*, Montreal, Canada, pp. 1753–1759, 2019.

[77] J. Jang, S. Ye, M. Seo. Can large language models truly understand prompts? A case study with negated prompts, [Online], Available: https://arxiv.org/abs/2209.12711, 2022.

[78] N. Poerner, U. Waltinger, H. Schütze. E-BERT: Efficient-yet-effective entity embeddings for BERT. In *Proceedings of the Findings of the Association for Computational Linguistics*, pp. 803–818, 2020. DOI: 10.18653/v1/2020.findings-emnlp.71.

[79] S. B. Li, X. G. Li, L. F. Shang, Z. H. Dong, C. J. Sun, B. Q. Liu, Z. Z. Ji, X. Jiang, Q. Liu. How pre-trained language models capture factual knowledge? A causal-inspired analysis. In *Proceedings of the Findings of the Association for Computational Linguistics*, Dublin, Ireland, pp. 1720–1732, 2022. DOI: 10.18653/v1/2022.findings-acl.136.

[80] B. Heinzerling, K. Inui. Language models as knowledge bases: On entity representations, storage capacity, and paraphrased queries. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 1772–1791, 2021. DOI: 10.18653/v1/2021.eacl-main.153.

[81] C. G. Wang, X. Liu, D. Song. Language models are open knowledge graphs, [Online], Available: https://arxiv.org/abs/2010.11967, 2020.

[82] S. Razniewski, A. Yates, N. Kassner, G. Weikum. Language models as or for knowledge bases, [Online], Available: https://arxiv.org/abs/2110.04888, 2021.

[83] B. AlKhamissi, M. Li, A. Celikyilmaz, M. Diab, M. Ghazvininejad. A review on language models as knowledge bases, [Online], Available: https://arxiv.org/abs/2204.06031, 2022.

[84] X. P. Qiu, T. X. Sun, Y. G. Xu, Y. F. Shao, N. Dai, X. J. Huang. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, vol. 63, no. 10, pp. 1872–1897, 2020. DOI: 10.1007/s11431-020-1647-3.

[85] T. Sun, X. Liu, X. Qiu, X. Huang. Raradigm shift in natural language processing. *Machine Intelligence Research*, vol. 19, no. 3, pp. 169–183, 2022.

[86] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, R. Lowe. Training language models to follow instructions with human feedback, [Online], Available: https://arxiv.org/abs/2203.02155, 2022.

[87] BigScience Workshop. BLOOM: A 176B-parameter open-access multilingual language model, [Online], Available: https://arxiv.org/abs/2211.05100, 2022.

[88] K. T. Song, X. Tan, T. Qin, J. F. Lu, T. Y. Liu. MASS: Masked sequence to sequence pre-training for language generation. In *Proceedings of the 36th International Conference on Machine Learning*, Long Beach, USA, pp. 5926–5936, 2019.

[89] S. Min, X. Lyu, A. Holtzman, M. Artetxe, M. Lewis, H. Hajishirzi, L. Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Abu Dhabi, UAE, pp. 11048–11064, 2022.

[90] Y. Goldberg. Assessing BERT's syntactic abilities, [Online], Available: https://arxiv.org/abs/1901.05287, 2019.

[91] A. Warstadt, Y. Cao, I. Grosu, W. Peng, H. Blix, Y. N. Nie, A. Alsop, S. Bordia, H. K. Liu, A. Parrish, S. F. Wang, J. Phang, A. Mohananey, P. M. Htut, P. Jeretic, S. R. Bowman. Investigating BERT's knowledge of language: Five analysis methods with NPIs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, Hong Kong, China, pp. 2877–2887, 2019. DOI: 10.18653/v1/D19-1286.

[92] E. Wallace, Y. Z. Wang, S. J. Li, S. Singh, M. Gardner. Do NLP models know numbers? Probing numeracy in embeddings. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, Hong Kong, China, pp. 5307–5315, 2019. DOI: 10.18653/v1/D19-1534.

[93] A. Ettinger. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 34–48, 2020. DOI: 10.1162/tacl_a_00298.

[94] Z. Bouraoui, J. Camacho-Collados, S. Schockaert. Inducing relational knowledge from BERT. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence, the 32nd Innovative Applications of Artificial Intelligence Conference, the 10th AAAI Symposium on Educational Advances in Artificial Intelligence*, New York, USA, pp. 7456–7463, 2020. DOI: 10.1609/aaai.v34i05.6242.

[95] X. H. Zhou, Y. Zhang, L. Y. Cui, D. D. Huang. Evaluating commonsense in pre-trained language models. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence, the 32nd Innovative Applications of Artificial Intelligence Conference, the 10th AAAI Symposium on Educational Advances in Artificial Intelligence*, New York, USA, pp. 9733–9740, 2020. DOI: 10.1609/aaai.v34i05.6523.

[96] A. Roberts, C. Raffel, N. Shazeer. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 5418–5426, 2020. DOI: 10.18653/v1/2020.emnlp-main.437.

[97] B. Y. Lin, S. Lee, R. Khanna, X. Ren. Birds have four

legs?! NumerSense: Probing numerical commonsense knowledge of pre-trained language models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 6862–6868, 2020. DOI: 10.18653/v1/2020.emnlp-main.557.

[98] A. Tamborrino, N. Pellicanò, B. Pannier, P. Voitot, L. Naudin. Pre-training is (almost) all you need: An application to commonsense reasoning. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 3878–3887, 2020. DOI: 10.18653/v1/2020.acl-main.357.

[99] A. Achille, M. Rovere, S. Soatto. Critical learning periods in deep networks. In *Proceedings of the 7th International Conference on Learning Representations*, New Orleans, USA, 2019.

[100] N. Saphra, A. Lopez. Understanding learning dynamics of language models with SVCCA. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, USA, pp. 3257–3267, 2019. DOI: 10.18653/v1/N19-1329.

[101] N. Saphra, A. Lopez. LSTMs compose—and learn—bottom-up. In *Proceedings of the Findings of the Association for Computational Linguistics*, pp. 2797–2809, 2020. DOI: 10.18653/v1/2020.findings-emnlp.252.

[102] S. Hochreiter, J. Schmidhuber. Long short-term memory. *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. DOI: 10.1162/neco.1997.9.8.1735.

[103] M. Raghu, J. Gilmer, J. Yosinski, J. Sohl-Dickstein. SVCCA: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Proceedings of the 31st Conference on Neural Information Processing Systems*, Long Beach, USA, pp. 6076–6085, 2017.

[104] Z. Z. Lan, M. D. Chen, S. Goodman, K. Gimpel, P. Sharma, R. Soricut. ALBERT: A lite BERT for self-supervised learning of language representations. In *Proceedings of the 8th International Conference on Learning Representations*, Addis Ababa, Ethiopia, 2020.

[105] T. Shen, Y. Mao, P. C. He, G. D. Long, A. Trischler, W. Z. Chen. Exploiting structured knowledge in text via graph-guided representation learning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 8980–8994, 2020. DOI: 10.18653/v1/2020.emnlp-main.722.

[106] I. Yamada, A. Asai, H. Shindo, H. Takeda, Y. Matsumoto. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 6442–6454, 2020. DOI: 10.18653/v1/2020.emnlp-main.523.

[107] T. Févry, L. B. Soares, N. FitzGerald, E. Choi, T. Kwiatkowski. Entities as experts: Sparse memory access with entity supervision. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 4937–4951, 2020. DOI: 10.18653/v1/2020.emnlp-main.400.

[108] L. Logeswaran, M. W. Chang, K. Lee, K. Toutanova, J. Devlin, H. Lee. Zero-shot entity linking by reading entity descriptions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, pp. 3449–3460, 2019. DOI: 10.18653/v1/P19-1335.

[109] D. Gillick, S. Kulkarni, L. Lansing, A. Presta, J. Baldridge, E. Ie, D. Garcia-Olano. Learning dense representations for entity retrieval. In *Proceedings of the 23rd Conference on Computational Natural Language Learning*, Hong Kong, China, pp. 528–537, 2019. DOI: 10.18653/v1/K19-1049.

[110] Y. J. Qin, Y. K. Lin, R. Takanobu, Z. Y. Liu, P. Li, H. Ji, M. L. Huang, M. S. Sun, J. Zhou. ERICA: Improving entity and relation understanding for pre-trained language models via contrastive learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pp. 3350–3363, 2021. DOI: 10.18653/v1/2021.acl-long.260.

[111] P. Banerjee and C. Baral. Self-supervised knowledge triplet learning for zero-shot question answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 151–162, 2020. DOI: 10.18653/v1/2020.emnlp-main.11.

[112] L. B. Soares, N. FitzGerald, J. Ling, T. Kwiatkowski. Matching the blanks: Distributional similarity for relation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, pp. 2895–2905, 2019. DOI: 10.18653/v1/P19-1279.

[113] V. Shwartz, P. West, R. Le Bras, C. Bhagavatula, Y. Choi. Unsupervised commonsense question answering with self-talk. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 4615–4629, 2020. DOI: 10.18653/v1/2020.emnlp-main.373.

[114] H. Tian, C. Gao, X. Y. Xiao, H. Liu, B. L. He, H. Wu, H. F. Wang, F. Wu. SKEP: Sentiment knowledge enhanced pre-training for sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4067–4076, 2020. DOI: 10.18653/v1/2020.acl-main.374.

[115] Y. Levine, B. Lenz, O. Dagan, O. Ram, D. Padnos, O. Sharir, S. Shalev-Shwartz, A. Shashua, Y. Shoham. SenseBERT: Driving some sense into BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4656–4667, 2020. DOI: 10.18653/v1/2020.acl-main.423.

[116] G. A. Miller. WordNet: A lexical database for English. In *Proceedings of a Workshop Held at Harriman: Speech and Natural Language*, New York, 1992.

[117] R. Navigli, S. P. Ponzetto. BabelNet: Building a very large multilingual semantic network. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, Uppsala, Sweden, pp. 216–225, 2010.

[118] J. Song, D. Liang, R. M. Li, Y. T. Li, S. R. Wang, M. L. Peng, W. Wu, Y. X. Yu. Improving semantic matching through dependency-enhanced pre-trained model with adaptive fusion. In: *Proceedings of the Findings of the Association for Computational Linguistics*, Abu Dhabi, UAE, pp. 45–57, 2022.

[119] K. Guu, K. Lee, Z. Tung, P. Pasupat, M. W. Chang. REALM: Retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, Article number 368, 2020.

[120] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. T. Yih, T. Rocktäschel, S. Riedel, D. Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, Curran Associates Inc, Vancouver, Canada, Article number. 793, 2020.

[121] M. Yasunaga, A. Aghajanyan, W. J. Shi, R. James, J. Leskovec, P. Liang, M. Lewis, L. Zettlemoyer, W. T. Yih. Retrieval-augmented multimodal language modeling, [Online], Available: https://arxiv.org/abs/2211.12561, 2022.

[122] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, USA, pp. 5998–6008, 2017.

[123] J. Wallat, J. Singh, A. Anand. BERTnesia: Investigating the capture and forgetting of knowledge in BERT. In *Proceedings of the 3rd BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pp. 174–183, 2020. DOI: 10.18653/v1/2020.blackboxnlp-1.17.

[124] A. Warstadt, A. Parrish, H. K. Liu, A. Mohananey, W. Peng, S. F. Wang, S. R. Bowman. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 377–392, 2020. DOI: 10.1162/tacl_a_00321.

[125] N. Kassner, P. Dufter, H. Schütze. 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*, pp. 3250–3258, 2021. DOI: 10.18653/v1/2021.eacl-main.284.

[126] S. C. Y. Chan, A. Santoro, A. K. Lampinen, J. X. Wang, A. Singh, P. H. Richemond, J. McClelland, F. Hill. Data distributional properties drive emergent in-context learning in transformers, [Online], Available: https://arxiv.org/abs/2205.05055, 2022.

[127] E. Voita, I. Titov. Information-theoretic probing with minimum description length. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 183–196, 2020. DOI: 10.18653/v1/2020.emnlp-main.14.

[128] A. Srivastava, A. Rastogi, A. Rao, A. A. Md Shoeb, A. Abid, A. Fisch, A. R. Brown, A. Santoro, A. Gupta, A. Garriga-Alonso, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models, [Online], Available: https://arxiv.org/abs/2206.04615, 2022.

[129] M. Hardt, E. Price, N. Srebro. Equality of opportunity in supervised learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, Barcelona, Spain, pp. 3323–3331, 2016.

[130] N. Kilbertus, M. Rojas-Carulla, G. Parascandolo, M. Hardt, D. Janzing, B. Schölkopf. Avoiding discrimination through causal reasoning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, USA, pp. 656–666, 2017.

[131] M. Kusner, J. Loftus, C. Russell, R. Silva. Counterfactual fairness. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Long Beach, USA, pp. 4069–4079, 2017.

[132] J. Vig, S. Gehrmann, Y. Belinkov, S. Qian, D. Nevo, Y. Singer, S. Shieber. Investigating gender bias in language models using causal mediation analysis. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, Vancouver, Canada, Article number 1039, 2020.

[133] A. Feder, K. A Keith, E. Manzoor, R. Pryzant, D. Sridhar, Z. Wood-Doughty, J. Eisenstein, J. Grimmer, R. Reichart, M. E. Roberts, B. M. Stewart, V. Veitch, D. Y. Yang. Causal inference in natural language processing: Estimation, prediction, interpretation and beyond. *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 1138–1158, 2022. DOI: 10.1162/tacl_a_00511.

[134] Y. Elazar, N. Kassner, S. Ravfogel, A. Feder, A. Ravichander, M. Mosbach, Y. Belinkov, H. Schütze, Y. Goldberg. Measuring causal effects of data statistics on language model's 'factual' predictions, [Online], Available: https://arxiv.org/abs/2207.14251, 2022.

[135] M. Finlayson, A. Mueller, S. Gehrmann, S. Shieber, T. Linzen, Y. Belinkov. Causal analysis of syntactic agreement mechanisms in neural language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pp. 1828–1843, 2021. DOI: 10.18653/v1/2021.acl-long.144.

[136] A. Köhn. What'S in an embedding? Analyzing word embeddings through multilingual evaluation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, pp. 2067–2073, 2015. DOI: 10.18653/v1/D15-1246.

[137] A. Gupta, G. Boleda, M. Baroni, S. Padó. Distributional vectors encode referential attributes. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, pp. 12–21, 2015. DOI: 10.18653/v1/D15-1002.

[138] Y. Yaghoobzadeh, K. Kann, T. J. Hazen, E. Agirre, H. Schütze. Probing for semantic classes: Diagnosing the meaning content of word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, pp. 5740–5753, 2019. DOI: 10.18653/v1/P19-1574.

[139] J. von Oswald, E. Niklasson, E. Randazzo, J. Sacramento, A. Mordvintsev, A. Zhmoginov, M. Vladymyrov. Transformers learn in-context by gradient descent, [Online], Available: https://arxiv.org/abs/2212.07677, 2022.

[140] A. Sinitsin, V. Plokhotnyuk, D. V. Pyrkin, S. Popov, A. Babenko. Editable neural networks. In *Proceedings of the 8th International Conference on Learning Representations*, Addis Ababa, Ethiopia, 2020.

[141] D. Ha, A. M. Dai, Q. V. Le. Hypernetworks. In *Proceedings of the 5th International Conference on Learning Representations*, Toulon, France, 2017.

[142] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, J. Taylor. Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, ACM, Vancouver, Canada, pp. 1247–1250, 2008. DOI: 10.1145/1376616.1376746.

[143] D. Vrandečić, M. Krötzsch. Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, vol. 57, no. 10, pp. 78–85, 2014. DOI: 10.1145/2629489.

[144] J. Pérez, M. Arenas, C. Gutierrez. Semantics and complexity of SPARQL. *ACM Transactions on Database Systems*, vol. 34, no. 3, Article number 16, 2009. DOI: 10.1145/1567274.1567278.

[145] T. Y. Gao, A. Fisch, D. Q. Chen. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pp. 3816–3830,

2021. DOI: 10.18653/v1/2021.acl-long.295.

[146] S. D. Hu, N. Ding, H. D. Wang, Z. Y. Liu, J. G. Wang, J. Z. Li, W. Wu, M. S. 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*, Dublin, Ireland, pp. 2225–2240, 2022. DOI: 10.18653/v1/2022.acl-long.158.

[147] T. X. Sun, Y. F. Shao, H. Qian, X. J. Huang, X. P. Qiu. Black-box tuning for language-model-as-a-service. In *Proceedings of the 39th International Conference on Machine Learning*, Baltimore, USA, pp. 20841–20855, 2022.

[148] K. Hambardzumyan, H. Khachatrian, J. May. WARP: Word-level Adversarial ReProgramming. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pp. 4921–4933, 2021. DOI: 10.18653/v1/2021.acl-long.381.

[149] B. Lester, R. Al-Rfou, N. Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Punta Cana, Dominican Republic, pp. 3045–3059, 2021. DOI: 10.18653/v1/2021.emnlp-main.243.

[150] G. H. Qin, J. Eisner. Learning how to ask: Querying LMs with mixtures of soft prompts. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5203–5212, 2021. DOI: 10.18653/v1/2021.naacl-main.410.

[151] X. Han, W. L. Zhao, N. Ding, Z. Y. Liu, M. S. Sun. PTR: Prompt tuning with rules for text classification. *AI Open*, vol. 3, pp. 182–192, 2022. DOI: 10.1016/j.aiopen.2022.11.003.

[152] B. Ozturkler, N. Malkin, Z. Wang, N. Jojic. Thinksum: Probabilistic reasoning over sets using large language models, [Online], Available: https://arxiv.org/abs/2210.01293, 2022.

[153] 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*, pp. 255–269, 2021. DOI: 10.18653/v1/2021.eacl-main.20.

[154] E. Ben-David, N. Oved, R. Reichart. PADA: A prompt-based autoregressive approach for adaptation to unseen domains, [Online], Available: https://arxiv.org/abs/2102.12206, 2021.

[155] T. Schick, S. Udupa, H. Schütze. Self-diagnosis and self-debiasing: A proposal for reducing corpus-based bias in NLP. *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 1408–1424, 2021. DOI: 10.1162/tacl_a_00434.

[156] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, Q. V. Le. Finetuned language models are zero-shot learners. In *Proceedings of the 10th International Conference on Learning Representations*, 2022.

[157] V. Sanh, A. Webson, C. Raffel, S. H. Bach, L. Sutawika, Z. Alyafeai, A. Chaffin, A. Stiegler, A. Raja, M. Dey, M. S. Bari, C. W. Xu, U. Thakker, S. S. Sharma, E. Szczechla, T. Kim, G. Chhablani, N. V. Nayak, D. Datta, J. Chang, M. T. J. Jiang, H. Wang, M. Manica, S. Shen, Z. X. Yong, H. Pandey, R. Bawden, T. Wang, T. Neeraj, J. Rozen, A. Sharma, A. Santilli, T. Fèvry, J. A. Fries, R. Teehan, T. L. Scao, S. Biderman, L. Gao, T. Wolf, A. M. Rush. Multitask prompted training enables zero-shot task generalization. In *Proceedings of the 10th International Conference on Learning Representations*, 2022.

[158] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Z. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Y. Dai, M. Suzgun, X. Y. Chen, A. Chowdhery, A. Castro-Ros, M. Pellat, K. Robinson, D. Valter, S. Narang, G. Mishra, A. Yu, V. Zhao, Y. P. Huang, A. Dai, H. K. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, J. Wei. Scaling instruction-finetuned language models, [Online], Available: https://arxiv.org/abs/2210.11416, 2022.

[159] S. S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Y. Chen, S. H. Chen, C. Dewan, M. Diab, X. Li, X. V. Lin, T. Mihaylov, M. Ott, S. Shleifer, K. Shuster, D. Simig, P. S. Koura, A. Sridhar, T. L. Wang, L. Zettlemoyer. OPT: Open pre-trained transformer language models, [Online], Available: https://arxiv.org/abs/2205.01068, 2022.

[160] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. S. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. C. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. W. Zhou, X. Z. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, N. Fiedel. PaLM: Scaling language modeling with pathways, [Online], Available: https://arxiv.org/abs/2204.02311, 2022.

[161] Q. X. Dong, L. Li, D. M. Dai, C. Zheng, Z. Y. Wu, B. B. Chang, X. Sun, J. J. Xu, L. Li, Z. F. Sui. A survey on in-context learning, [Online], Available: https://arxiv.org/abs/2301.00234, 2023.

[162] D. H. Lee, A. Kadakia, K. M. Tan, M. Agarwal, X. Y. Feng, T. Shibuya, R. Mitani, T. Sekiya, J. Pujara, X. Ren. Good examples make a faster learner: Simple demonstration-based learning for low-resource NER. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, Dublin, Ireland, pp. 2687–2700, 2022. DOI: 10.18653/v1/2022.acl-long.192.

[163] J. Eisenstein, D. Andor, B. Bohnet, M. Collins, D. Mimno. Honest students from untrusted teachers: Learning an interpretable question-answering pipeline from a pretrained language model, [Online], Available: https://arxiv.org/abs/2210.02498, 2022.

[164] H. X. Zhang, Y. Z. Zhang, R. Y. Zhang, D. Y. Yang. Robustness of demonstration-based learning under limited data scenario. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Abu Dhabi, UAE, pp. 1769–1782, 2022.

[165] S. Y. Li, J. S. Chen, Y. L. Shen, Z. Y. Chen, X. L. Zhang, Z. K. Li, H. Wang, J. Qian, B. L. Peng, Y. Mao, W. H. Chen, X. F. Yan. Explanations from large language models make small reasoners better, [Online], Available: https://arxiv.org/abs/2210.06726, 2022.

[166] Z. Y. Dai, V. Y Zhao, J. Ma, Y. Luan, J. M. Ni, J. Lu, A. Bakalov, K. Guu, K. B. Hall, M. W. Chang. Promptagat-

or: Few-shot dense retrieval from 8 examples, [Online], Available: https://arxiv.org/abs/2209.11755, 2022.

[167] W. H. Yu, D. Iter, S. H. Wang, Y. C. Xu, M. X. Ju, S. Sanyal, C. G. Zhu, M. Zeng, M. Jiang. Generate rather than retrieve: Large language models are strong context generators, [Online], Available: https://arxiv.org/abs/2209.10063, 2022.

[168] J. Wei, X. Z. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, D. Zhou. Chain-of-thought prompting elicits reasoning in large language models, [Online], Available: https://arxiv.org/abs/2201.11903, 2022.

[169] A. K. Lampinen, I. Dasgupta, S. C. Y. Chan, K. Mathewson, M. H. Tessler, A. Creswell, J. L. McClelland, J. X. Wang, F. Hill. Can language models learn from explanations in context? In *Proceedings of the Findings of the Association for Computational Linguistics*, Abu Dhabi, UAE, pp. 537–563, 2022.

[170] D. Zhou, N. Schärli, L. Hou, J. Wei, N. Scales, X. Z. Wang, D. Schuurmans, O. Bousquet, Q. Le, E. Chi. Least-to-most prompting enables complex reasoning in large language models, [Online], Available: https://arxiv.org/abs/2205.10625, 2022.

[171] J. C. Liu, D. H. Shen, Y. Z. Zhang, B. Dolan, L. Carin, W. Z. Chen. What makes good in-context examples for GPT-3? In *Proceedings of the Deep Learning Inside Out: The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, Dublin, Ireland, pp. 100–114, 2022. DOI: 10.18653/v1/2022.deelio-1.10.

[172] O. Rubin, J. Herzig, J. Berant. Learning to retrieve prompts for in-context learning. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Seattle, USA, pp. 2655–2671, 2022. DOI: 10.18653/v1/2022.naacl-main.191.

[173] H. J. Su, J. Kasai, C. H. Wu, W. J. Shi, T. L. Wang, J. Y. Xin, R. Zhang, M. Ostendorf, L. Zettlemoyer, N. A. Smith, T. Yu. Selective annotation makes language models better few-shot learners, [Online], Available: https://arxiv.org/abs/2209.01975, 2022.

Boxi Cao received the B. Sc. degree in Beijing University of Posts and Telecommunication, China in 2019. He is a Ph. D. degree candidate at the Chinese Information Processing Laboratory, Institute of Software, Chinese Academy of Sciences, under the supervision of Professor Xianpei Han and Professor Le Sun.

His research interests lie in natural language process, especially the knowledge in large language models, as well as information extraction.

E-mail: boxi2020@iscas.ac.cn
ORCID iD: 0000-0001-9916-7406

Hongyu Lin received the Ph. D. degree from Institute of Software, Chinese Academy of Sciences, China in 2020. He is currently an associate professor in Institute of Software, Chinese Academy of Sciences, China.

His research interests include information extraction and knowledge mechanism in large LMs.

E-mail: hongyu@iscas.ac.cn

Xianpei Han received the Ph. D. degree in pattern recognition and intelligent systems from National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China in 2010. He is a professor of computer science at the Chinese Information Processing Laboratory, Institute of Software, Chinese Academy of Sciences, China.

His research interests lie in natural language understanding, and he has published about 60 papers in ACL/EMNLP/SIGIR/AAAI.

E-mail: xianpei@iscas.ac.cn (Corresponding author)
ORCID iD: 0000-0002-1304-6302

Le Sun received the Ph. D. degree in engineering mechanics from Nanjing University of Science Technology, China in 1998. He is a professor at Institute of Software, Chinese Academy of Sciences (ISCAS), China. He is the General Secretary of Chinese Information Processing Society of China (CIPS), China. He visited University of Birmingham, UK and University of Montreal, Canada as a visiting scholar at 2004 and 2005. He has published more than 100 top journal and conference papers. He received the best short paper Award from SIGIR2021. In 2022, he received the Excellent Tutor Award from Chinese Academy of Sciences and First prize of Qian Weichang Chinese Information Processing Science and Technology Award.

His research interests include natural language understanding, knowledge graph, information extraction, and question answer.

E-mail: sunle@iscas.ac.cn