Large Language Models and Knowledge Graphs: Opportunities and Challenges

Jeff Z. Pan
The University of Edinburgh, United Kingdom

Simon Razniewski
Bosch Center for AI, Germany

Jan-Christoph Kalo
Vrije Universiteit Amsterdam, The Netherlands

Sneha Singhania
Max Planck Institute for Informatics, Germany

Jiaoyan Chen
The University of Manchester & University of Oxford, United Kingdom

Stefan Dietze
GESIS - Leibniz Institute for the Social Sciences & Heinrich-Heine-Universität Düsseldorf, Germany

Hajira Jabeen
GESIS - Leibniz Institute for the Social Sciences, Germany

Janna Omeliyanenko
University of Würzburg, Germany

Wen Zhang
Zhejiang University, China

Matteo Lissandrini
Aalborg University, Denmark

Russa Biswas
Hasso-Plattner Institute, Germany

Gerard de Melo
Hasso-Plattner Institute, Germany

Angela Bonifati
Lyon 1 University, France

Edlira Vakaj
Birmingham City University, United Kingdom

Mauro Dragoni
Fondazione Bruno Kessler, Italy

Damien Graux
Edinburgh Research Centre, CSI, Huawei Technologies UK, United Kingdom

Abstract

Large Language Models (LLMs) have taken Knowledge Representation—and the world—by storm. This inflection point marks a shift from explicit knowledge representation to a renewed focus on the hybrid representation of both explicit knowledge and parametric knowledge. In this position paper, we will discuss some of the common debate points within the community on LLMs (parametric knowledge) and Knowledge Graphs (explicit knowledge) and speculate on opportunities and visions that the renewed focus brings, as well as related research topics and challenges.

…, Vol. 000, Issue 111, Article No. 42, pp. 42:1–42:30
…
…

2012 ACM Subject Classification General and reference → General literature, General and reference
Keywords and Phrases Large Language Model, Pre-trained Language Model, Knowledge Graph, Ontology, Retrieval Augmented Language Models
Digital Object Identifier 10.1234/0000000.00000000
Received Date of submission **Accepted** Date of acceptance **Published** Date of publishing

Editor Editor Name

1 Introduction

Large Language Models (LLMs) have taken Knowledge Representation (KR)—and the world—by storm, as they have demonstrated human-level performance on a vast spectrum of natural language tasks, including some tasks requiring human knowledge. Following this, people are gradually starting to accept the possibility of having knowledge represented in the parameters of some language models. The arrival of LLMs announces the era of Knowledge Computing, in which the notion of reasoning within KR is broadened to many computation tasks based on various knowledge representations.

This is a big step for the field of Knowledge Representation. For a long time, people focused on explicit knowledge, such as those embedded in texts, sometimes also known as unstructured data, and those in a structured form, such as in databases and knowledge graphs (KGs) [123]. Historically, for a long time, humans used texts to pass down their knowledge from one generation to another, until around the 1960s, when researchers started to study knowledge representation for better natural language understanding and developed early systems, such as ELIZA [180] at the MIT. In the early 2000s, the Knowledge Representation and the Semantic Web communities worked together to standardize the widely used knowledge representation languages, such as RDF [121] and OWL [55], at web scale, using which the large-scale knowledge bases are then more widely known as KGs [123], due to their helpful graph structures, enabling the both logical reasoning and graph-based learning.

This inflection point, with the arrival of LLMs, marks a paradigm shift from explicit knowledge representation to a renewed focus on the hybrid representation of both explicit knowledge and parametric knowledge. As a popular approach for explicit knowledge representation, KGs are now widely investigated for the combination with Transformer-based LLMs, including pre-trained masked language models (PLMs) like BERT [39] and RoBERTa [104], and more recent generative LLMs like the GPT series [23] and LLaMA [165]. Some works use LLMs to augment KGs for, e.g., knowledge extraction, KG construction, and refinement, while others use KGs to augment LLMs for, e.g., training and prompt learning, or knowledge augmentation. In this paper, considering both directions, LLMs for KGs and KGs for LLMs, we present a better understanding of the shift from explicit knowledge representation to a renewed focus on the hybrid representation of both explicit knowledge and parametric knowledge.

A related survey paper [204] presents a comprehensive review of using LLMs for KG construction and reasoning, while our work provides a more in-depth view of the inflection point, considering not only relational KGs but also KGs with ontologies as schemas, as well as other dimensions of structured knowledge, including tabular data [183] and numerical values [122]. Other works on the intersection of LLMs and KGs have a minor overlap with the topics covered in our paper; e.g., on studies using LLMs as KGs [5], on using KGs to augment LLMs [185], or on comparing GPT-4 with ChatGPT and SOTA fine-tuning methods on three knowledge-related tasks—entity, relation and event extraction, link prediction, and KG question answering [204]. Overall, none of these papers look into the implications of the inflection point with concrete applications. To this end, this paper summarises the common debate points within the community, introduces the state-of-the-art for a comprehensive set of topics where KGs and LLMs are integrated, and further presents opportunities and challenges.

2 Common Debate Points within the Community

The usage of parametric and explicit knowledge together is a topic of debate in the Knowledge Computing community, with proponents and skeptics offering different perspectives. Below are some summaries of common points of contention.

Knowledge Representation and Reasoning: KGs offer a structured representation of knowledge with explicit relationships, enabling reasoning and inference [110]. Critics argue that parametric knowledge in LLMs relies on statistical patterns rather than true understanding and reasoning [13]. Proponents of LLMs like ChatGPT, highlight their ability to generalize from large-scale text corpora, capturing a range of information, and excellent language understanding capabilities. On the one hand, LLMs could generate plausible but incorrect or nonsensical responses, such as hallucinations, due to a lack of explicit knowledge representation [193]. There are also doubts on whether LLMs have the ability to learn directional entailments [96] or infer subsumption between concepts [61]. On the other hand, KGs can be costly to build. While LLMs can be expensive to train too, they can be readily usable to support many downstream applications, bringing AI from the backstage to centre stage. Thus parametric knowledge is not the (only) destination for LLMs. To sum up, in comparison to the classic trade-off between expressiveness and decidability in Knowledge Representation, here we have the trade-off between precision and recall considering using explicit and parametric knowledge in Knowledge Computing tasks.

High Precision Methods: The success of KGs can largely be attributed to their ability to provide factual information about entities with high accuracy. For instance, YAGO [154] asserts an accuracy rate exceeding 95%. Similarly, Google necessitates high accuracy in its KG for operational use, e.g., the semi-automatical construction method of Knowledge Vault was not utilized in production, partly due to its inability to reach the desired 99% accuracy in their benchmark [179]. Along this line of thought, many LLM-based methodologies for KG completion fail to reach these high-performance levels, as exemplified by the performance of BERT in [97, 167], and GPT-3, equipped with hundreds of billions of parameters [4]. This calls for novel high precision methods for KG construction based on LLMs.

Numerical Values: It is widely recognized that LLMs grapple with handling numerical values. Even straightforward arithmetic tasks can be a struggle for LMs, as highlighted in a study by Big-bench [153]. This shortcoming also extends to KG completion tasks [78]. Multiple LLMs have been evaluated on their ability to complete KGs using numerical facts from Wikidata [169], such as individuals' birth and death years. However, none of the tested models accurately predicted even a single year. This raises questions about the capability of current LLMs to correctly memorize numbers during pre-training in a way that enables them for subsequent use in KG completion. While LLMs like PaLM [153] demonstrate some proficiency in dealing with numbers, more commonly used smaller models seem ill-equipped for this task. The complexity escalates when considering the intricacies of metrics and diverse numbering formats and types. Currently, modifying LLMs to handle numerical values remains unresolved, making their utilization for numerical KG completion seem far from practical.

Long-tail Knowledge: One of the key research questions on LLMs for the Knowledge Computing community (and beyond) is how much knowledge LLMs remember [107]. Investigations indicate that LLMs' performance significantly deteriorates when dealing with random Wikidata facts, specifically those associated with long-tail entities, in comparison to popular entities, as evidenced in the PopQA dataset [107] and other datasets [133, 167]. This effect can be traced back to a causal relationship between the frequency of an entity's appearance in the pre-training corpus and the LLMs' capacity for memorization [44]. Even sizable LLMs face difficulties when trying to retain information about long-tail entities [80]. KGs inherently present an advantage over LLMs through their provision of knowledge about long-tail entities [78, 167] and thus can further help improve the recall for Knowledge Computing tasks.

Bias, Fairness and Beyond: Critics argue that LLMs can perpetuate and amplify biases present in the training data, leading to biased outputs. LLMs may generate biased or prejudiced responses if the training data contains stereotypes or discriminatory information [113, 91]. On the other hand, proponents argue that bias is not inherent to LLMs but reflects societal biases embedded in the data. They emphasize the importance of addressing bias in the training data and developing mitigation techniques [144, 134, 140]. A survey [16] argued that "bias" in Natural Language Processing (NLP) systems took various conceptualizations without being critically engaged by practitioners. KGs are also used in a plethora of downstream tasks, and social biases engraved in KG embeddings get propagated [56, 87]. Specifically, ontology creation, which generally comprises manual rules factored by opinions, motivations, and personal choices, is a source of bias [73, 43]. Also, automated pipelines for KG construction exhibit gender bias [109]. There are other similar concerns of LLMs beyond bias and fairness, including (but not limited to) copyright violation and misinformation. In general, due to the implicit nature of parametric knowledge, it is less straight forward to forget such toxic information from LLMs, compared to explicit knowledge.

Explainability and Interpretability: KGs are often preferred in scenarios where explainability and interpretability are crucial [28], as they explicitly represent relationships between entities and provide a structured knowledge representation. Skeptics of LLMs argue that these models lack transparency and interpretability, making it difficult to understand how they arrive at their answers or recommendations. Proponents of LLMs acknowledge the challenge of explainability but argue that recent research efforts [8, 72] are improving LLM's interpretability through techniques like attention mechanisms, model introspection. Some also argue that Chain-of-Thoughts (CoT) [177] can also improve the explainability of LLMs, although question decomposition and precisely answering sub-questions with LLMs are still far from being solved. Attribution evaluation and augmentation of LLMs with e.g., source paragraphs and sentences is another recent research topic for improving their explainability in question answering [17].

3 Opportunities and Visions

One of the key questions this paper needs to answer is, now with the emergence of parametric knowledge, what new opportunities do we have? Here are some of our thoughts on such new opportunities with the arrival of parametric knowledge and its potential integration with explicit knowledge.

1. **Instant access to huge text corpora:** As mentioned in the Introduction, for a long time, human beings passed down their knowledge in texts. Thus, a lot of knowledge these days are in textual form. Using LLMs gives access to extremely large text corpora at high speed, and recently even on consumer hardware [65]. This allows AI developers to avoid getting bogged down in previously critical challenges around data gathering, preparation, storage, and querying at scale. It also helps to reduce previously critical dependencies on the field of information retrieval.
2. **Richer knowledge for many subtasks:** Although the most prominent capabilities of LLMs, question answering and dialogue, are still under critical scrutiny, it should not be overlooked

that LLMs have significantly advanced and simplified many traditional tasks of the knowledge engineering pipeline. Out-of-the-box, with fine-tuning on a few examples, or via few-shot prompting, LLMs have advanced many tasks such as dependency and structured parsing, entity recognition, and relation extraction. And just as errors propagate along a pipeline, so do improvements, thus enabling KG construction at unprecedented scale and quality. Furthermore, LLMs are readily usable for many downstream tasks beyond knowledge engineering. By injecting explicit, and in particular structured, knowledge into LLMs, such as through retrieval augmented methods, one can make explicit knowledge more readily usable for such wide range of downstream tasks, further realising the vision of 'Knowledge is power'.

3. **Even more advanced language understanding:** LLMs alone already significantly advanced the "understanding" of natural language, as evidenced by tasks like textual entailment, summarization, paraphrase detection and generation, etc. These capabilities are critical to making knowledge engineering robust to linguistic variance, typographic errors, redundancy, and other features of human-written, web-scraped, and other noisy forms of text. Now with potential novel approaches to combining parametric knowledge with explicit knowledge, it is possible to have even more advanced language understanding, not only for textual entailments, but also for other NLP tasks, such as summarization and consistent generation.
4. **Compression entails consolidation:** An important step in traditional knowledge engineering is the consolidation and aggregation of conflicting and concurring pieces of information, requiring often elaborate methods for consolidating observations from sentences, patterns, and constraints [149]. In LLM training, an aggregation occurs automatically. Although this step is not entirely understood, it brings the potential for outsourcing a major challenge in knowledge engineering.

With the above new opportunities brought by the combination of parametric and explicit knowledge, our vision is two-folded:

- In **Explicit-Knowledge-First** use cases, our vision is that *LLMs will enable, advance, and simplify crucial steps in the knowledge engineering pipeline so much as to enable KGs at unprecedented scale, quality, and utility.*
- In **Parametric-Knowledge-First** use cases, our vision is that *KGs will improve, ground, and verify LLM generations so as to significantly increase reliability and trust in LLM usage.*

Both visions are neither alternatives, nor does one build upon the other. Instead, we believe that classes of use cases will continue to exist side-by-side, some of which favor Explicit-Knowledge-First (scrutable) approaches, some of which favor Parametric-Knowledge-First (blackbox) approaches, with either of these having significant potential for benefiting from synergies of the two directions.

4 Key Research Topics and Related Challenges

With the opportunities and visions related to the availability of both parametric and explicit knowledge in place, in this section, we categorize, summarize, and present the recent developments in using LLMs and KGs under four different themes.

4.1 LLMs for KGs: Knowledge Extraction and Canonicalisation

KG construction is a complex task that demands collecting and integrating information from a wide array of sources, encompassing structured, semi-structured, and unstructured data. Traditional methods often rely on specific modules designed to process each data type in isolation and struggle when the content is diverse and heterogeneous in structure. However, LLMs are powerful NLP models trained on a broad spectrum of information sources, making them well-suited for knowledge extraction tasks. This section presents work that uses LLMs for knowledge extraction from various sources.

4.1.1 Entity Resolution and Matching

Entity resolution (also known as entity matching, entity linking or entity alignment) is the process of linking pieces of information occurring in multiple heterogeneous datasets and referring to the same world entity [46, 50, 126]. Past research has focused on developing the methodologies and similarity measures among entities represented by flat structured data. However, entity resolution for semi-structured data for KGs is a fairly recent topic with significantly less attention. Approaches for entity alignment can be split into general vs embedding-based categories.

General approaches, such as CG-MuAlign [203] which employs Graph Neural Networks (GNNs) to perform multi-type entity alignment, leverages the neighborhood information and generalizes to unlabeled types, and REA [129] which tackles the multi-lingual entity alignment problem by combining adversarial training with GNNs to cope with the problem of noisy labeled data provided as input.

Embedding-based entity alignment methods for KGs reduces the symbolic similarities between graph entities to a vector space in order to flatten the heterogeneity of graph components and facilitate reasoning [156]. Specifically, a total of 23 representative embedding alignment approaches are cross-compared in terms of performance but also are shown to require significant supervision in the labeling phase. Therefore, unsupervised methods and methods that cope with large-scale KGs are highly desirable in future research investigations.

LLMs are used in entity resolution and linking for KGs in multiple ways [7]. First, LLMs can help with labeling training data, which is typically a resource-intensive and time-consuming step, hindering the performance of entity alignment for KGs. Similar to how [146] employs Generative Adversarial Networks (GANs) to reduce the effort of labeling data, we argue that LLMs can provide labeled samples of KGs and control the performances of the aforementioned embedding-based approaches. Also, LLMs can help build solid corpora of entity matching rules, modulo the fact that a declarative formalism with a logical language L is defined in the graph setting. Training data for this logical language should be provided as input to LLMs, similarly to SQL statements that are readily available for consumption in text corpora. However, prompt engineering is needed in order to produce meaningful rule corpora for real-world large-scale KGs, such as DBpedia [9] and Wikidata [169]. Entity matching rule logs can be envisioned for these real-world large-scale KGs in a similar fashion as query logs for these KGs [18, 19].

Concluding, entity alignment and matching are necessary pre-processing steps for full-fledged knowledge reasoning. The combination of general entity linking approaches with embedding-based ones, as well as the leveraging of LLM-driven rule and labeled data construction, can lead to better integration of LLMs with knowledge reasoning [66]. The latter integration of LLMs and knowledge reasoning can also boost performance, thus enabling the interpretability and explainability of the model's output and filling the gap between symbolic and statistical AI.

4.1.2 Knowledge Extraction from Tabular Data

Extracting knowledge from tabular data like databases, Web tables and CSV files is a common way for KG construction. For tables whose semantics (meta information) are already known, heuristic rules can be defined and used to transform their data into KG facts. However, real-world tables often have unclear semantics with important meta information, such as table name and column header, not clearly defined. Meanwhile, the original data usually need to be retrieved, explored, integrated and curated, before expected knowledge can be extracted. In recent years, Transformer-based LMs have been investigated for processing tables especially their textual contents. They can be applied to table vector representation as a foundation of other prediction tasks [168]. TURL [38] is a typical method of table representation learning that uses BERT [39] and has been applied in several tasks such as cell filling, column type annotation, and relation extraction. Similarly, RPT [162] uses BERT and GPT to pre-train a table representation model. Starmie [47] transforms columns into sequences using a template and fine-tunes BERT with a contrast learning framework using unionable and not unionable column pairs as samples.

Among all the table processing tasks, semantic table annotation which matches table data to KG components (e.g., table column to KG class, table cell to KG entity, inter-column relationship to KG property) can be directly applied to extract knowledge for KG construction and population [103, 76]. There have been several attempts that use LLMs for these tasks. Doduo [155] serializes a table into a sequence of tokens and trains BERT for predicting column types and inter-column relationships. Korini et al. [86] prompts ChatGPT to annotate semantic column types. When task-specific samples for demonstration are minimal or absent, ChatGPT achieves similar performance RoBERTa model. Although some attention has been given to utilizing LLMs for tabular data processing and KG construction, there is still much room for investigation, especially towards the following challenges:

- **Transforming table contents into sequences:** A table or a table element with its structured context needs to be transformed into a sequence before it can be fed into LLMs. Different transformation methods are required for different LLM utilization scenarios, such as fine-tuning LLMs, LLM inference with prompts, and instruction tuning of LLMs.
- **Representing and utilizing none textual tabular data:** A table often contains not only long and short text but also data of other types like numbers and dates. There are still few works that consider these data.
- **Extracting tabular knowledge:** LLMs are mostly applied to process and understand tables but rarely applied to the final step of knowledge extraction. OntoGPT [25], which extracts instances from texts to populate an ontology using ChatGPT is known, but there are no counterparts for tables. Beyond instances, extracting relational facts is more challenging.

4.1.3 Knowledge Extraction from Text

Knowledge extraction from text generally entails automatic extraction of entities and associated relations, with traditional pipelines processing vast amounts of sentences and documents. This process enables the transformation of raw text into actionable knowledge, facilitating various applications such as information retrieval, recommendation systems, and KG construction. The language understanding capabilities of LLMs have enhanced this process.

For example, (1) Named Entity Recognition (NER) and Entity Linking, as mentioned in Section 4.1.1, involve identifying and categorizing named entities (such as persons, organizations, and locations) in text and linking (more in Section 4.2.1) them to KGs. (2) Relation extraction focuses on identifying and classifying relationships between entities, with LLMs that leverage zero-shot and few-shot in-context learning techniques [178, 93]. (3) Event extraction aims to detect and classify events mentioned in the text, including their participants and attributes [170, 194]. (4) Semantic Role Labeling (SRL) involves identifying the roles played by entities in a sentence, such as the subject, object and predicate [148, 199].

These approaches allow LLMs to extract information from text without needing a large amount of explicit training on specific domains, thereby improving their versatility and adaptability.

Moreover, LLMs have demonstrated proficiency in extracting knowledge from languages other than English, including low-resource languages, paving the way for cross-lingual knowledge extraction and enabling the utilization of LLMs in diverse linguistic contexts [89].

Furthermore, prompting LLMs has introduced new paradigms and possibilities in the field of NLP. LLMs can generate high-quality synthetic data, which can then be used to fine-tune smaller task-specific models. This approach, known as synthetic data generation, addresses the challenge of limited training data availability and enhances the performance of models [77, 163]. Additionally, instruction tuning has emerged as a powerful technique where LLMs are trained on datasets described by explicit instructions, enabling more precise control and customization of their behavior for specific tasks [178, 174]. Also, for constructing domain-specific KGs, the stakes are higher, and hence scrutinizing the generated text (by experts) is necessary. However, it is still a step forward since human annotation is less expensive than human text generation.

Apart from the obvious substantial computational resource requirements for training and utilizing these LLM, there are various challenges, including the ones presented in Section 2. More specifically, the following future directions are still possible:

- **Efficient extraction from very long documents.** Current LLMs are incapable of processing very long documents like novels in one shot. In this regard, modeling long-range dependencies and performing corpus-level information extraction can be further improved.
- **High-coverage information extraction.** Almost all extraction pipelines focus on high precision. However, high recall is ignored or under-explored [152]. Building knowledge extractors with high precision and high recall will result in a great leap toward building lifelong information extractors.

4.2 LLMs for KGs: Knowledge Graph Construction

We highlight the important role that LLMs play in improving KG construction, focusing on current trends, issues, and unanswered questions in this field. We start by discussing link prediction, which is a way to generate new facts given an existing KG. Next, we look at inductive link prediction, a method that predicts triples for unseen relationships. Our focus then shifts to a more recent method where triples is directly extracted from the parametric knowledge of an LLM.

As a conclusion of this section, we discuss the challenges of LLM-based methods for KG construction. These involve issues with long-tail entities, numerical values, and also the precision of these methods.

4.2.1 Link Prediction

Link prediction is about predicting a missing element of a triple given the other two elements. It includes head entity prediction $(?, r, t)$, relation prediction $(h, ?, t)$, and tail prediction $(h, r, ?)$.

KG link prediction methods have mostly been studied for static snapshots of a KG. Many approaches, in fact, assume a training phase in which the current state of the KG is used to learn embeddings and other model parameters. Subsequently, many such models can only operate on entities for which an embedding was learned in the training phase. This leaves them incapable of predicting links for any previously unseen entities, such as newly added people or products. Inductive link prediction (ILP), in contrast, focuses on techniques that can predict links to new entities not originally contained in a KG. Furthermore, existing KG embedding-based KG completion approaches frequently fail to leverage textual information and other literal information [54].

To overcome these challenges, current research focuses on incorporating textual information available in the KGs to improve the KG embeddings, boost performance in downstream tasks, and to support ILP. The latent representation is learned from textual information using a variety of encoding models such as linear models, convolutional models, recurrent neural models, and LLMs and studied in [105, 124]. In this work, we focus only on LLM-based methods.

The LLM encoder BERT [39] is used in Pretrain-KGE [197] to generate initial entity embeddings from entity descriptions and relations. These embeddings are then fed into KG embedding models to generate final embeddings. MADLINK [14] uses SBERT to generate embeddings from entity descriptions, together with the entity embeddings obtained from structural information. KEPLER [175] offers a unified approach for KG embedding and pre-trained language representation, embedding text-enhanced knowledge and factual knowledge into LLMs. Nayyeri et al. [114] use LLMs to produce representations at word, sentence, and document levels, merging them with graph structure embeddings. Huang et al. [69] propose a model that combines LLMs with other modalities, such as vision, to generate a multi-model embedding space. CoDEx [3] uses a novel loss function driven by LLMs that helps KG embedding models estimate the likelihood of triples based on textual information. While these approaches can leverage the structural information contained in the graph, KG embeddings can not be directly used to predict unknown entities in the ILP setting. To still predict entities within ILP, existing works that combine text embeddings and traditional KG embeddings use only the text embeddings in the ILP setting [106] or apply similarity based heuristics to generate KG embeddings for unseen entities [171].

Instead of considering the graph structure, another line of research directly leverages LLMs for KG Completion. For example, KG-BERT [187] represents a triple as an ordered sequence of head text, including surface form, descriptions, and attributes, relation text, tail text separated with special tokens. KG-BERT [187] optimizes the BERT model on KGs, followed by KG-GPT2 [15] that fine-tunes GPT-2 model. MTL-KGC [84] enhances the effectiveness of KG-BERT by combining prediction and relevance ranking tasks. PKGC evaluates triple validity by translating it into natural language sentences, while LLMs process these sentences for binary classification. Masked Language Model (MLM) is introduced to encode KG text, with MEMKGC [32] predicting masked entities using the MEM classification model. Open world KGC [33] expands MEMKGC with modules Entity Description Prediction (EDP) and Incomplete Triple Prediction (ITP), focusing on predicting entities with given textual descriptions. StAR [172] uses Siamese-style textual encoders for text and a scoring module, while SimKGC [173] leverages a Siamese textual encoder. LP-BERT [94] is a hybrid KG completion method that combines MLM encoding for pre-training with LLM and separated encoding for fine-tuning, using a contrastive learning strategy. Also, LLMs such as GPT-3 and GPT-4 have the in-context learning capability which could adapt to new tasks through careful prompt design without fine-tuning the model parameters. Since LLMs are directly capable of predicting entities that are not yet contained in the KG, many works using direct LLM approaches also evaluate their models in the ILP setting [32, 33, 36, 172, 173, 175, 192].

Beyond the described approaches that construct and leverage embeddings for link prediction, LLMs may also be directly used in a prompting setting to find suitable links between entities, with existing methods described in detail in the following Section 4.2.2. As with the previous approaches that purely use LLMs, prompt-based approaches are directly applicable for ILP and are commonly evaluated in this setting [20, 74, 133, 150, 200].

Challenges and Opportunities: LLMs make it significantly easier to jointly utilize structural and text information for link prediction, while there are still challenges:

- The generative language model which uses a decoder-only or encoder-decoder architecture cannot ensure that the generated result is already included in the KG. Also, one entity may have multiple natural language names. Thus it is hard to judge the correctness of the generated results from LLMs.

- Current link prediction models are mostly evaluated on ranking metrics, such as Hit@k and Mean Reciprocal Rank, which requires methods to give a ranked list of candidates. Considering that the number of candidate entities might be huge (e.g., over ten thousand), framing each candidate as an sample requires too many times of predictions and LLMs can not be efficiently evaluated due to its large scales (e.g., over 100 billion parameters) and high computation cost.
- The key research question of link prediction is how well a method could learn to infer new triples based on existing ones. LLMs are trained based on a massive corpus that might overlap with KGs such as Wikidata [169]. Thus it is not easy to distinguish whether the LLM completes the prediction by utilizing its memory or reasoning over existing facts.
- LLMs provide several benefits for ILP as demonstrated by the many existing approaches, but their application also has certain drawbacks. Since ILP entities may occur that are not contained within the underlying KG, the task is inherently reliant on auxiliary information. When leveraging LLMs for ILP, many studies focus on improving the extraction of knowledge from LLMs through prompt engineering, which is, by itself, a current popular research area that may directly benefit the domain.
- While prompting provides promising results, it requires well-designed strategies for predicting multi-token entities and out-of-vocabulary tokens. Further, if required knowledge is not captured by the LLM, e.g., when querying novel concepts that emerged after LLM pre-training, schemes to incorporate further knowledge into the LLM are required. Meanwhile, as feature selection, finding a suitable prompt also needs much searching with many experiments, which is costly especially for those recent commercial LLMs like GPT-4.
- Alternative methods rely on available high-quality textual descriptions of unknown entities. These methods inherit the limitations of classical link prediction, in that they require one-vs-all comparisons against all entity candidates during inference, which may become computationally prohibitive for large KGs. As such, efficient strategies for obtaining predictions may provide a promising research direction.
- Further, the combination of these methods with classical link prediction models has already shown strong performance and may be further explored to incorporate structural information into LLM-driven approaches.

Though challenges exist, opportunities lie in designing efficient and effective link prediction methods combining LLMs preserving the efficiency of traditional methods and robustness of language models.

4.2.2 LLMs for KGs: Triple Extraction from LLMs

Traditionally, retrieval and reasoning of relational knowledge have both relied on symbolic knowledge bases [51], that often are constructed using supervised extraction techniques applied to unstructured corpora, e.g. Web archives [190, 164]. More recently, self-supervised LLMs have been investigated for their ability to directly retrieve relational knowledge [202] from their parameters, e.g. through question answering, prompting through the use of cloze-style questions [62, 143] or statement scoring [157]. In this context, the ability of LLMs to retrieve, infer and generalize relational knowledge is seen as a crucial indicator of their capacity to understand and interpret natural language. Even though a range of terms are used in that context, e.g. fact or knowledge retrieval as well as knowledge inference, we refer to the task of accessing relational knowledge from LLM parameters as *knowledge retrieval*.

Benchmarks and Baselines: LAMA is the first benchmark dataset to evaluate knowledge retrieval in LLMs [132]. Related works show that knowledge retrieval through prompts is inconsistent with regard to paraphrasing [45, 62], with some types of information guiding LLMs towards more correct answers [24, 131, 31], while other are harmful to their performance [125, 83].

LLMs struggle to retrieve knowledge from low-frequency phenomena [141] and [70] argue that LLMs fail to express large varieties of knowledge when prompted for it in a zero-shot manner.

Zhong et al. [200] propose that the models' accuracy may be from memorizing training data, not actually inferring knowledge. Similar to LAMA, the experiments on a more recent probing work KAMEL [78] confirm that LLMs are still far from the knowledge access capabilities of symbolic knowledge bases. The Knowledge Memorization, Identification and Reasoning test work KMIR [53] reveals that while LLMs struggle to robustly recall facts, their capacity to retain information is determined more by the number of parameters than the training methods, and while model compression can help preserve the memorization performance, it reduces the ability to identify and reason about the information in LLMs from transformer-based language models. Linzbach et al. [98] also present similar findings. LLMs are known to struggle with more complex reasoning tasks [68, 61]. Branco et al. [21] explore generalisability of common-sense reasoning capabilities and the impact of shortcuts in training data.

Biases in Triple Extraction Evaluation: LLMs may exhibit various types of biases; representation of the majority viewpoint being a common issue due to distributions prevalent within pretraining data [12], neglecting disagreements among multiple viewpoints (e.g. by majority voting) [35]. Prior works investigate individual factors (such as frequency) or LLM biases in other tasks [108], as well as knowledge retrieval [200].

With respect to the interpretation, reliability and generalisability of knowledge retrieval, several studies [21, 24] investigate whether LLMs actually learn transferable generalisations or only exploit incidental shortcuts in the data. [24] explore biases in three different knowledge retrieval paradigms, namely *prompt-based retrieval*, *case-based analogy*, *context-based inference*, finding that decent performance of existing knowledge retrieval baselines tends to be driven by biased prompts that overfit to artefacts in the data, guide the LLM towards correct entity types or unintentionally leak correct answers or additional constraints applicable to the correct answer. In a similar context, [42] discuss the shortcut learning behaviour arising due to skewed training datasets, the model, or the fine-tuning process. [145] demonstrate an intriguing similarity between human cognitive biases and those exhibited by LLMs. Using insights from psychology, they analyse the learning and decision-making processes of black-box models to reveal their biases towards right-and-wrong for decision-making. Therefore, rigorous assessment of existing benchmark datasets is necessary for generalizable insights about knowledge retrieval and inference performance, and to facilitate efficient, unbiased knowledge retrieval from LLMs.

Prompt Engineering for Triple Extraction: Cao et al. [24] propose three paradigms for factual knowledge extraction from LLMs: prompt-based, case-based, and context-based. Results suggest the prompt-based retrieval is biased towards prompt structure. Prompt engineering [10] aims to create prompts that efficiently elicit desired responses from LLMs for a specific task. However, a limited number of manually created prompts only reveal a portion of the model's encoded knowledge [74], as the response can be influenced by the phrasing of the question. Thus, prompt engineering is a crucial part of knowledge retrieval from LLMs. LPAQA [74] uses an automated mining-based and paraphrasing-based method to generate diverse high-quality prompts, as well as ensemble methods to combine answers from different prompts. Automatic Prompt Engineer [202] uses LLM models like InstructGPT [119] and instruction induction [64] to generate instruction candidates which are then improved by proposing semantically similar instruction variants to achieve human-level performance. Zhou et al. [202] investigate the ability of LLMs, such as GPT-3, to generate high-quality prompts for a variety of tasks.Initial experiments on the role of syntax in knowledge retrieval [98] find a strong dependency on prompt structure and knowledge retrieval performance.

To summarise, prior works have shown that relational knowledge is captured by LLMs to a certain extent. However, there is still insufficient understanding of how performance differs across different kinds of knowledge or relations, for instance, commonsense knowledge compared to entity-centric encyclopedic facts or transversal versus hierarchical relations. In addition, several studies raise questions about LLMs capacity to infer knowledge beyond already-seen statements.

Challenges and Opportunities:

- **Entity Disambiguation:** Entity disambiguation is essential for KG construction to ensure unique identification of entities and to maintain consistency across the graph. However, when extracting facts from LLMs, entity disambiguation presents several challenges, since LLMs only operate on word token level. Hence, polysemy and homonymy make it difficult to determine the correct entity when a term has multiple meanings or is spelled the same as others but has different meanings. Also, the need to resolve co-references, where the same entity is mentioned in various ways within a text, further complicates the process. Moreover, the same piece of text can refer to different entities depending on the context, making it a significant challenge to correctly identify and classify the entities. Entities that were not present in the training data or are less common in general can be particularly hard to disambiguate. This can be a frequent issue with newer, less well-known, or very domain-specific entities. These complexities lead to major challenges that need to be addressed: enhancing disambiguation techniques to better handle long-tail entities; developing methods to better understand and utilize context in entity disambiguation; and improving co-reference resolution in a way that it can be effectively incorporated into KG construction.
- **Long-tail Entities:** Existing LLMs still manifest a low level of precision on long-tail entities. Models may begin to generate incorrect information when they fail to memorize the right facts. The answers provided by these models often lack consistency. Incorrect correlations drawn from the pre-training corpus can lead to various biases in KG completion. Whether retrieval-augmented models serve as a viable solution to this problem remains uncertain, as does the potential necessity to adapt pre-training and fine-tuning processes to enhance model robustness in handling long-tail entities.
- **High-Precision:** LLMs face challenges in achieving high-precision predictions when performing knowledge extraction [167]. A potential strategy to derive high-precision KGs from LLMs is to focus on model calibration. However, there are pressing challenges that remain unsolved. How can LLM training be adapted to prioritize high-precision learning? Can LLMs be used for validation purposes? These questions form the crux of the ongoing exploration in this field.
- **Provenance:** Extracting factual knowledge directly from LLMs does not provide provenance, the origin and credibility of the information, which presents multiple issues. Without provenance, verifying the accuracy of information becomes challenging, potentially leading to the spread of misinformation. Additionally, bias detection is hindered, as the lack of source information makes it difficult to account for potential biases in the data used for training. Provenance also provides critical context, without which, information can be misunderstood or misapplied. Lastly, the absence of source information compromises model transparency, making it hard to evaluate the accountability of the LLMs.

4.3 LLMs for KGs: Ontological Schema Construction

A KG is often equipped with an ontological schema (including rules, constraints and ontologies) for ensuring quality, enabling easier knowledge access, supporting reasoning, etc. Meanwhile, an independent ontology, which usually represents conceptual knowledge sometimes with logics, can also be regarded as a KG. In this part, we introduce topics that LLMs are applied to learn ontological schemas and to manage ontologies.

4.3.1 Constraint and Rule Mining from KGs

The existing KGs are mostly obtained from large-scale data extraction pipelines, which are notoriously brittle and can introduce errors and inconsistencies in these graphs [40, 137]. Moreover, a KG is *never considered complete* since the closed world assumption does not hold [40, 128], i.e., it is not possible to conclude that a missing fact is false *unless it contradicts another existing fact.* Instead, we usually consider that in a KG it holds the open-world assumption, that is a missing fact is simply considered as *unknown.*

Practical applications impose high demands in terms of (semi-)automatic methods for data quality assessment and validation [85, 136, 2]. Since KGs contain huge amounts of data, it is not feasible to manually inspect and correct their errors. Therefore, a common approach is to instantiate rules and constraints that can be automatically enforced. These constraints express dependencies and conditions that the KG needs to satisfy at all times and that should not be violated by the introduction of new facts or their deletion. In KGs, rules and constraints can take the form of Graph Functional Dependencies [48], declarative first-order logic rules [52], or *validating shapes* [85, 135]. Once a set of rules or constraints are instantiated, the next step is to either identify which entities or facts in the KG violate any of them, or employ them to delete erroneous information, or, finally, to employ them to deduce any missing information [49, 138].

▶ **Example 1.** The following rules could apply for a subset of a graph describing people and their relationships:

$$r_1 : \texttt{hasParent}(x, y) \Rightarrow \texttt{hasChild}(y, x);$$
$$r_2 : \texttt{hasParent}(x, y) \Rightarrow \exists v_1, v_2 \in \mathbb{N} \mid \texttt{hasBirthYear}(x, v_1) \wedge \texttt{hasBirthYear}(y, v_2);$$
$$r_3 : \texttt{hasChild}(x, y) \wedge \texttt{hasBirthYear}(x, v_1) \wedge \texttt{hasBirthYear}(y, v_2) \Rightarrow v_1 < v_2;$$

Where r_1 states that `hasChild` is the inverse equivalent relation of `hasParent`, r_2 states that for each person in the KGs for which we know the parent-child relationship we should know the birth year, and r_3 states that if y is a child of x then x should be born before y.

Nonetheless, a fundamental challenge is how to generate such rules and constraints. Specifying them manually is prohibitively difficult and expensive [2, 136]. On the one hand, the domain experts, who know the semantics for the dataset at hand, may not have the skill set or the background necessary to formally express those rules. Even when skilled, domain experts would require a substantial amount of manual work to exhaustively materialize a complete list of such rules [137]. Therefore, in the past decade, we have witnessed an increased interest in methods that can (semi-) automatically extract rules and constrains [137, 196]. In all those cases, the data in the KG is exploited as a *silver standard* [128]. That is, while the data are not perfect, we can still assume it is at least of reasonable quality. This approach works for rules that can find frequent positive examples. The challenge arises for rules that need to govern entities and facts that are rare, and for negative rules that need to prohibit some facts from appearing. To handle these cases, contextual information can be exploited. In this context, LLMs bring a number of important opportunities:

- ability to extract contextual information from input text;
- information extraction from contextual data used during training;
- automatic generation of new rules via inductive reasoning;

- understanding vocabulary in context, allowing to reconcile rules with synonyms and polysemy;
- providing explanations of rules or generating candidate validating or counterfactual examples.

Consider, for example, a rule where the relationships `hasParent` has been used, an LLM can be interrogated to identify whether `hasParent` between two entities refers more likely to the semantic meaning of being biologically descended or being owned or controlled. LLMs can be used to filter out candidate constraints, e.g., given their ability to mimic simple common sense reasoning, rules can be translated in the form of questions to the LLM, e.g., "*if X is child of Y is it true that the birth year of Y comes always before the birth year of X?*". Thus, an LLM can be used in support or replacement of a human annotator towards generated rules.

One of the most promising abilities of LLMs is parsing long texts. In companies and organizations, documents exist that contain reference governing information, e.g., procedures, regulations, and specifications. Here we see an untapped opportunity in parsing these documents in relation to the entities and predicates in the KG to extract constraints. Yet, the challenge arises in the fact that the LLM needs to use the correct vocabulary of entities and relations and the correct rule syntax. Recent advancements in program synthesis show a possible way to the challenge.

Finally, an even more fundamental challenge is that of transcending the usage of LLMs for NLP alone, and using them directly on large sets of facts within a KG. We can think of this setting as a special kind of multi-modal LLMs, where the KG is a specific modality. This brings to the question of whether it is possible to train LLMs to treat the task of rule generation as, for example, a summarization task. This would require then the ability to perform both inductive and abductive reasoning and treat rules as summaries from the set of facts in the KG.

A Rule Mining Case with ChatGPT Let's have a look at the following real example asking ChatGPT to generate rules.

▶ **Example 2.** We ask ChatGPT "Please list three rules that could help to infer one person's father. And output it in a first-order rule form." The following is the output we get:

> "Certainly! Here are three rules expressed in first-order logic form that could help infer a person's father:
>
> Rule 1: $\forall x, y\ (\texttt{child}(x, y) \land \texttt{male}(y) \rightarrow \texttt{father}(y, x))$ This rule states that for all individuals x and y, if there exists a 'child' relationship between x and y and y is male, then y is the father of x.
>
> Rule 2: $\forall x, y\ (\texttt{biologicalChild}(x, y) \rightarrow \texttt{father}(y, x))$ This rule states that for all individuals x and y, if there exists a 'biologicalChild' relationship between x and y, then y is the father of x.
>
> Rule 3: $\forall x, y\ (\texttt{fatherOf}(y, x) \rightarrow \texttt{father}(y, x))$ This rule states that for all individuals x and y, if there exists a 'fatherOf' relationship between y and x, then y is the father of x.
>
> These rules, expressed in first-order logic, demonstrate how the relationships between individuals can be used to infer the father of a person within a KG."

From the above example, we can see that ChatGPT gives three rules that can possibly infer the father relationship. Rule 1 is correct which is consistent with the definition of father; Rule 2 is a rule with high probability; and Rule 3 is also correct since `father` and `fatherOf` are inverse relations. This demonstrates LLMs have rule knowledge and could express it in a formal form following instruction, thus they could serve as experts for automatic rule construction. To this end, however, we need to **align the rules from the LLM to the existing knowledge in the KG**, and **feed the LLM about the existing knowledge and the context of the KG**. Both are consistent with discussion about the company and family examples.

4.3.2 Ontology Refinement

Ontology refinement includes quite a few topics like knowledge completion (e.g., subsumption prediction, complex concept learning and new concept placement), erroneous knowledge detection and repair (e.g., inconsistency checking) and knowledge canonicalization (e.g., entity renaming). Besides formally represented knowledge, real-world ontologies, such as the widely used medical ontology SNOMED CT[1] and food ontology FoodOn[2], also include a lot of meta information defined by different annotation properties for usability, such as entity labels, synonyms and natural language definition. Taking the concept *obo:FOODON_00002809* in FoodOn as an example, it has not only formal knowledge such as named super concepts and logical restrictions, but also labels and synonyms (e.g., "edamame"), definitions (e.g., "Edamame is a preparation of immature soybean ..."), comments and so on. These meta information, especially the natural language text, further motivates people to use LLMs for ontology refinement.

For a refinement task, usually there are quite a few existing examples in the original ontology. Therefore, a straightforward solution, which has been adopted by most current methods, is fine-tuning a Pre-trained Language Model such as BERT together with an attached classifier. One typical method is BERTSubs [26] which is to predict the subsumption relationship between two named concepts, or between one named concept and one complex concept. It concatenates the corresponding texts of the two candidate concepts with special tokens as the input of a model composed of a pre-trained BERT and a logistic regression classifier, and fine-tunes the model with the existing subsumptions in the target ontology. For a named concept, the text could be either its name (label or synonym), or its name in combination with a textual description of its surrounding concepts; while for a complex concept, the text is its description (a.k.a. verbalisation). Another typical work is [101] which fine-tunes BERT and predicts the position to place in SNOMED CT for a new concept. Note that there are also some language model-based methods in taxonomy curation, such as [147] which fine-tunes BERT for taxonomy edge completion and GenTaxo [191] which fine-tunes a BERT variant named SciBERT for predicting positions that need new concepts. They can be directly applied or easily extended to refine an ontology's concept hierarchies.

Exploiting LLMs is a promising direction for ontology refinement, but it still needs much effort before they become practical tools. DeepOnto [59], which is a Python-based package that can support quite a few ontology engineering tasks, has already included some tools for ontology refinement and alignment using LLMs, but more development is needed to make it more accessible and to support generative LLMs like LLaMA and GPT-4. One obvious challenge is that those very recent generative LLMs have been rarely explored for ontology engineering. However, we think the following two research challenges are more fundamental:

- **Exploiting the graph structure and logics of an ontology together with its text**. Currently LLM fine-tuning-based methods can well utilize the text of individual entities, but their other formal semantics are often ignored or not effectively incorporated. Besides fine-tuning with samples constructed by some templates, more LLM techniques such as prompt learning and instruction tuning could be considered.
- **Combing symbolic reasoning with LLM inference**. Symbolic reasoning, such as consistency checking in OWL ontologies, can still play a role to e.g., validate the knowledge inferred by LLMs. One aspect to incorporate symbolic reasoning is constructing samples for LLM fine-tuning and extracting prompts for LLM inference, while another aspect is the synergized framework [204] where LLM inference and symbolic reasoning work iteratively.

[1] `https://www.snomed.org/`
[2] `https://foodon.org/`

4.3.3 Ontology Alignment

The content of one single ontology is often incomplete and many real-world applications rely on cross-domain knowledge. Ontology alignment (a.k.a. ontology matching), which is to identify cross-ontology mappings between entities that have an equivalent, subsumption or membership relationship, thus becomes especially important for knowledge integration. The entity can be a concept (class), an individual (instance) or a property. Traditional systems (e.g., LogMap [75]) heavily rely on lexical matching and optionally use symbolic reasoning to remove mappings that lead to logical conflicts; while some recent methods combine these techniques with machine learning techniques like feature engineering, semantic embedding and distant supervision for better performance (e.g., LogMap-ML [27]). Especially, when the ontologies have a large ratio of assertions (large ABoxes) and the task is to discover equivalent individuals, ontology alignment is very close to the KG entity alignment task that has been widely investigated in recent years using KG embeddings [198]. As in ontology refinement, exploiting the textual information by applying LLMs is a promising direction for augmenting ontology alignment.

The study of LLM application in ontology alignment is similar to ontology refinement. Pre-trained language models such as BERT have been applied via fine-tuning [115, 58]. BERTMap [58] is a typical system that has achieved state-of-the-art performance in many biomedical ontology alignment benchmarks. It fine-tunes a pre-trained LM with synonym pairs extracted from the original ontologies and the potentially given mappings, and combines the predicted concept equivalence scores with lexical matching scores and reasoning for the mappings. Those recent LLMs like GPT-4 have not been applied in ontology alignment, as far as we know, and the two fundamental research challenges mentioned in ontology refinement are applicable in ontology alignment. Besides, ontology alignment has another critical challenge:

- **Evaluating LLM-based ontology alignment systems**. Novel evaluation protocols with new metrics are needed to fairly and efficiently compare LLM-based systems even with incomplete ground truth mappings [60]. Meanwhile, the semantics from the textual meta information and the LLM may be inconsistent with formal semantics defined in ontologies, and thus it is sometimes hard to determine whether a mapping by an LLM-based system is true or not.

4.4 KGs for LLMs: Training and Accessing LLMs

In Sections 4.1 to 4.3, we discussed on three different aspects on using LLMs for KGs. In this section, we look into the other direction, i.e., using KGs for LLMs. There are a few dimensions here. Firstly, KGs can be used as training data for LLMs. Secondly, triples in KGs can be used for prompt construction. Last but not least, KGs can be used as external knowledge in retrieval augmented language models.

4.4.1 KGs for (Pre-)Training Language Models

KGs usually contain information extracted from highly trusted sources, post-processed, and vetted by human evaluations. Information from KGs has been integrated into the pre-training corpus since natural language text alone can lead to limited information coverage [187, 130, 1, 184].

Using factual knowledge from KGs to pre-train LLMs has also infused structured knowledge [112]. This integration of KGs with LLMs, along with efficient prompts, has made it convenient to inject world knowledge and incorporate new evolving information into language models [41]. Additionally, knowledge expressed in high-resource language KBs has been transferred into LMs tuned for low-resource languages [201, 100].

Furthermore, grounding knowledge from KGs to pre-train LMs has shown improvements in performance on generation and QA tasks [30, 142, 120]. In another approach, [166] proposed an interpretable neuro-symbolic KB, where the memory consists of vector representations of entities and relations from an existing KB. These representations are augmented to an LM during pre-training and fine-tuning, enabling the model to excel in knowledge-intensive QA tasks.

4.4.2 KGs for Prompt Construction

The attention received by the integration of KGs and LLMs has grown recently. On the one hand, there is the explored direction of prompting LLMs for collecting and distilling knowledge in order to make it available to the end-users. On the other hand, there is the less explored research direction where KGs are used in synergy with prompts in order to enhance LLMs with capabilities making them more effective and, at the same time, trustworthy. A number of studies have leveraged KGs to enrich and fine-tune prompt creation resulting in a significant increase in prompt quantity, quality, and diversity compared to manual approaches. KGs have been employed in single and in multi-turn scaffolding prompts at scale, powered by numerous traversal paths over KGs with low authoring cost while considering the meaningful learning patterns [90]. Other studies have investigated how incorporating explicit knowledge from external sources like KGs can help prompt engineering, especially by giving additional contexts (e.g., attributes, K-hop neighbors) of the entities in order to help the LLMs to generate better predictions [22]. Approaches like KnowPrompt [31] use KGs to incorporate semantic and prior knowledge among relation labels into prompt-tuning for relation extraction, enhancing the prompt construction process and optimizing their representation with structured constraints. Certain studies have utilized LLMs and prompts in the task of reasoning over KGs [34], e.g., LARK uses entities and relations in queries to find pertinent sub-graph contexts within abstract KGs, and then, performs chain reasoning over these contexts using LLM prompts of decomposed logical queries outperforming previous state-of-the-art approaches by a significant margin.

Challenges and Opportunities: The current research in the field of KG utilization for prompt creation predominantly centers around LLMs, which are considered to have relatively lower efficacy compared to LLMs. LLMs present significant potential for advancing prompt creation methodologies in conjunction with KGs. We may summarize this perspective within the following four challenges:

- **C1:** KGs can be employed to automatically extract and represent relevant knowledge to generate context-aware writing prompts. Analyze and understand the relationships between different writing prompts, enabling the generation of prompts that build upon each other.
- **C2:** KGs can be combined with LLMs to facilitate the interactive and dynamic generation of prompts, adapting to user feedback and preferences in real time. Furthermore, the use of KGs in prompt creation has opened up possibilities for explainability and interpretability. Since KGs provide a transparent representation of knowledge, the prompts generated from KGs can be easily traced back to their underlying sources.
- **C3:** KGs can integrate into prompts the definitions of guards exploited during the generative task. Such guards may lead to enhancing the trustworthiness of the information generated by LLMs and make them more compliant with specific domain-wise or context-wise constraints.
- **C4:** KGs can create prompts that ask questions (e.g., inferring missing relations in an incomplete KG) that trigger KG complex reasoning capabilities and intermediate reasoning steps.

The integration of KGs within the prompt construction activities will allow us to answer the following preparatory set of research questions. For each research question, we provide a link to the challenges mentioned above, aiming to identify appropriate research pathways.

- **RQ1:** How can KGs be integrated into existing prompts to enhance the effectiveness (including relevance, non-biased, and privacy-preserving) of the information extracted from LLMs? $\longrightarrow$ C1, C2, C3.
- **RQ2:** How can KGs be exploited to drive the generative capabilities of LLMs in order to properly address the whole ethical constraints of AI-based solutions? $\longrightarrow$ C2, C3.
- **RQ3:** What are the optimal approaches to generate KGs-based prompts that enhance reasoning capabilities? $\longrightarrow$ C4.

Also, in the following three scenarios, the community would benefit from tackling the above-mentioned challenges and research questions.

1. **KGs for Hallucination Detection in LLMs:** The reliability of LLMs is greatly affected by the hallucination problem, where they generate inaccurate information. Despite attempts to address it, the issue of hallucination is likely to persist in the realm of LLMs for the foreseeable future. To aid in the detection of hallucinations, KGs-based prompting aims to offer reliable information that can serve as a foundation. By combining LLMs and KGs, researchers can develop a comprehensive prompt-based fact-checking model that can identify hallucinations in various domains.
2. **KGs for Editing Knowledge in LLMs:** LLMs possess the ability to store extensive real-world knowledge, but they struggle to exploit prompts to update their internal knowledge to reflect real-world changes. KGs-based prompts offer a potential solution for modifying knowledge in LLMs, but they are restricted to handling basic tuple-based knowledge in KGs. Indeed, even if the entire LLM undergoes re-training, the knowledge presented through prompts would likely be assimilated within the vast network structure of the LLM.
3. **KGs for Black-box LLMs Knowledge Injection:** While pre-training and knowledge editing methods can bring LLMs up to date with the latest information, they require access to the internal structures and parameters of LLMs. However, many cutting-edge large LLMs only offer APIs that allow users and developers to interact with them, keeping their internal workings hidden from the public. Consequently, traditional KG injection techniques that involve modifying LLM structures with additional knowledge fusion modules cannot be employed. One potential solution is to convert various types of knowledge into different text prompts. However, it remains an area of ongoing research to determine if these prompts can effectively adapt to new LLMs. Additionally, the approach of using KGs-based prompts is constrained by the length of input tokens accepted by LLMs. Therefore, the question of how to enable effective knowledge injection for black-box LLMs still remains unanswered.

4.4.3 Retrieval Augmented Methods

There are a few of reasons that retrieval augmented methods are necessary for LLMs to obtain external knowledge. One reason is to address the problem of knowledge cutoff, i.e., LLMs are not aware of the events that happened after their training. Also, although parametric knowledge would increase when the size of parameters increases, training LLMs is expensive; e.g., GPT-3 (175B parameters) costs $4.6 million to train, and PaLM (540B parameters) costs $17 million. In fact, research suggests that the obtained knowledge from such training is mainly about popular entities [107]. Furthermore, for domain specific applications, there might be some significant knowledge that is not yet in LLMs, including private and business critical knowledge that cannot be put into LLMs.

One idea to deal with the above lack of (updated) knowledge is to edit the knowledge in LLMs. A obvious strategy is to retrain and fine-tune the model based on the modified data. However, apart from being costly, retraining cannot guarantee that erroneous data will be corrected. Another Strategy is to develop hyper-network to learn a parameter shift for the base model. De Cao et al. [37] trained a hyper-network, KnowledgeEditor, to modify a fact and used Kullback-Leibler (KL) divergence-constrained optimization to alleviate the side effect on other data/knowledge that should not be changed. However, this method does not perform well when editing *multiple edits*, as it uses the same strategy to process multiple edits and ignore the relation between different edit gradients, resulting in a "zero-sum" phenomenon, where the inter-gradient conflict will inevitably cause some data modifications to fail. Han et al. [57] design explicit and implicit multi-editor models to learn diverse editing strategies in terms of dynamic structure and dynamic parameters respectively, allowing to deal with the conflict data in an efficient end-to-end manner.

However, the above Knowledge Editing methods are not yet scalable, people started to introduce retrieve-generate architectures for building retrieval augmented generation models. These methods are mainly using unstructured passages as external knowledge. RAG [92] outperforms DPR [82] by marginalizing the retrieval step to train the generator and retriever jointly with the supervision of the label answer. FiD [71] encodes the concatenation of the passages retrieved by pre-trained DPR and the original question separately, and then fuses them with concatenation to the decoder. It is expected that structured knowledge will be the main source of external knowledge, as passages often contain noise. Knowledge Graphs can be used directly as external knowledge. They can also be used to enhance passage-based methods [189].

Retrieval augmentation is a very promising direction. There are a few pressing challenges:

- **Unifying Knowledge Editing and Retrieval Augmentation**: KGs can be used for editing knowledge in LLMs, while at the same time, KGs can also be used as external knowledge to assist LLMs in retrieval augmented methods. In fact, knowledge editing and retrieval augmentation is getting very close. For example, Mitchel et al. [111] proposed a Retrieval-Augmented Counterfactual Model (SERAC), which stores edits in an explicit memory for knowledge editing over LLMs.
- **Semi-parametric LLMs**: This direction is highly related to the topic of this position paper. The idea is to make use of explicit knowledge to augment LLMs. One of the key issue is to integrate different explicit knowledge [158], including unstructured ones, such as passages, and structured ones, such as KGs and databases, for augmenting LLMs.
- **Support of Complex Reasoning**: Can we go beyond simply retrieving explicit knowledge by enabling reasoning through retrieval augmented methods? BehnamGhader et al. [11] demonstrated with their experimental results that the similarity metric used by the retrievers is generally insufficient for reasoning tasks. Furthermore, LLMs do not take the complicated relations between statements into account, thus leading to to poor reasoning performance.

4.5 Applications

The integration of KGs and LLMs in a unified approach holds significant potential, as their combination mutually enhances and complements each other in a valuable manner. For instance, KGs provide very accurate and explicit knowledge, which is crucial for some applications i.e. healthcare, whereas LLMs have been criticized for their lack of factual knowledge leading to hallucinations and inaccurate facts. secondly, LLMs lack explainability instead, KGs given their symbolic reasoning ability, are able to generate interpretable results. On the other hand, KGs are difficult to construct from unstructured text and suffer from incompleteness therefore, LLMs could be utilized in addressing these challenges by text processing. Various applications have adopted this methodology of combining LLMs with KGs, such as healthcare assistants[3], question answering systems [188] or ChatBots, and sustainability, among others.

4.5.1 Commonsense Knowledge

The majority of KGs capture facts of the sort one might encounter in an encyclopedia or in a relational database. However, commonsense knowledge is another important form of world knowledge for AI systems. For instance, we may wish for a KG to not only capture that the Congo rainforest lies in Central Africa, but also that tropical rainforests have significant rainfall and lush green vegetation. ConceptNet is the most well-known commonsense knowledge graph, developed using manual crowdsourcing along with automated refinement techniques [102]. However, crowdsourcing is very labor-intensive and costly, so alternative means of harvesting such knowledge have long been sought.

Commonsense Knowledge from LLMs: The first study to investigate extracting knowledge from a language model to the best of our knowledge was indeed one that targeted commonsense knowledge [159]. The authors mined commonsense triples such as `hasProperty (apples, green)` from the Google Web 1T n-gram data as well as from Microsoft's Web-scale smoothed language models [67]. This was later extended into a large-scale commonsense knowledge graph [161] that covered a range of different relations and became a part of the WebChild KG [160].

As both crowdsourcing and information extraction from text are likely to lead to incomplete knowledge, a key challenge is how to generalize beyond what has been collected. The WebBrain project explored neural knowledge graph completion [29] for better generalization. COMET, short for COMmonsEnse Transformer [20], and the improved COMET-ATOMIC 2020 [70], used existing data to fine-tune Transformer-based models. This line of work considers the original ConceptNet relations as well as reasoning-related knowledge pertaining to events, causes, and effects, e.g., what goals might have motivated a person A to leave an event without person B. Recently, [181] shows how common-sense triples could be extracted from an LLM and use through distillation to transfer knowledge into a smaller LM, outperfoming the larger one. Overall, fine-tuned LLMs are found to outperform off-the-shelf LLMs, while also benefitting from the advances of the latter.

Challenges and Opportunities: Commonsense knowledge, in particular, is genuinely open-ended, such that it depends on a number of considerations whether it makes sense to attempt to materialize relevant knowledge beforehand or rather invoke a (possibly slow) LLM on the fly. Commonsense knowledge may also differ substantially between different cultures [116]. This also leads to the question of what kinds of biases are acceptable. Finally, a long-term challenge is how to capture knowledge that is not easily expressed in language, e.g., how a robot ought to grasp different kinds of objects.

4.5.2 Digital Build Environment

In the domain of Build Environment, where it is vital to design and construct in a safe and sustainable way, a number of regulations and guidelines need to be met. Automated Compliance Code Checking has bloomed to support this, but still, in the past, it was quite challenging to interpret regulations and execute rules in 3D models automatically due to the ambiguity of the text, and the need for extensive expert knowledge for interpretation. Many applications

[3] https://neo4j.com/blog/doctor-ai-a-voice-chatbot-for-healthcare-powered-by-neo4j-and-aws/

now in this area have combined LLMs + KGs in addressing these challenges by following an LLMs enhanced KG approach [204]. LLMs interpret the text in the regulations and enhance a KG of rules, which are further serialized using dome Domain Specific Language [195, 176]. Interrogating 3D models modeled as graphs using the Linked Building Data approach is another challenge for the domain as it requires skill sets in query languages like SPARQL. LLMs are helping in understanding human language written questions and converting those into relevant query languages by bringing a new way of how domain experts interact and interrogate 3D models and their various forms. An example of this is AI Speckle [4].

4.5.3 Digital Healthcare

The Digital Healthcare sector holds immense potential for various possibilities concerning the adoption of LLMs, including the automation of clinical documentation, the synthesis of patient histories, and the identification of potential candidates for clinical trials. Although these advancements are remarkable, it is crucial to recognize the potential risks associated with employing LLMs in healthcare. Indeed, Digital Healthcare is one of the most critical application domains for the adoption of LLMs. The needs of the major stakeholders (i.e., physicians, healthcare providers, and policymakers) row against the paradigm behind the creation of LLMs. In particular, the two major significant risks related to the model's accuracy and the privacy concerns stemming from its usage.

Accuracy. Some demonstrations of LLMs have showcased impressive capabilities. Nevertheless, there have also been documented instances where LLMs have made mistakes or exhibited erratic behavior. In the Digital Healthcare sector, where patient safety is of utmost importance, it is crucial for healthcare organizations to comprehend the potential risks associated with LLM usage. When utilized to diagnose hypothetical patient cases, LLMs have exhibited accuracy at a level comparable to that of a third- or fourth-year medical student, albeit not reaching a professional's proficiency. Despite this high level of performance, LLMs have also been known to generate false information, invent sources, commit logical errors, and provide answers that are inappropriate or unethical. The integration of KGs would definitely enhance the capabilities of LLMs given the possibility of injecting domain-specific knowledge able to mitigate the issues mentioned above. Avoiding hallucinations and preserving ethics are definitely the two major aspects to which LLMs+KG may contribute in a significant manner.

Privacy. One major concern with LLMs is that employing any third-party application necessitates the transmission of data to that party. When data, including protected health information (PHI), is managed by a covered entity like a hospital, it becomes subject to the regulations of the jurisdiction where the entity is located (e.g., GDPR). Furthermore, by sending PHI to additional third parties, organizations lose control over how that data will be handled. For instance, healthcare organizations are unable to determine the exact storage location of their data, whether it will be mixed with data from other organizations and utilized to train future language models, or what security measures are in place to safeguard the data. Healthcare organizations employing LLMs must recognize that their data are potentially more susceptible to breaches or misuse. The role of preserving private information may be played by KGs through the modeling of axioms defining which data may be shared and with who and how personal knowledge may be anonymized in order to be transmitted to possible external systems.

An alternative approach, prioritizing privacy, involves running an open-source LLM within the infrastructure of a healthcare organization. This way, it would be possible to directly work

[4] https://speckle.systems/blog/ai-powered-conversations-with-speckle/

on the parameters of LLMs given the possibility of having control of the injected knowledge and, at the same time, ensuring that data are never shared with a third party. However, these open-source models are currently not as advanced or extensively trained as the more popular commercial systems (e.g., ChatGPT). Additionally, the effort to create the knowledge resources to inject and the expertise required to program and maintain an open-source LLM may not be readily available to many healthcare organizations.

The two risks described above lead to likewise challenges that must be tackled in order to make a significant step toward the adoption of these models within the clinical practice.

4.5.4 Domain Specific Content Search

Recently we have witnessed the success of models like GPT-4 [118] in a multitude of NLP applications that involve multiple modalities and domain specific adaptations. While LLMs are often treated as generative models, they can be easily adopted in search and reasoning tasks in many tools and pipelines [5]. Nonetheless, as seen in the past all these specialized domains can better benefit from the inclusion of symbolic knowledge in machine methods [6]. Here we focus on two distinct applications: (1) semantic image and video search and (2) technical document understanding.

Recently, many methods, primarily based on deep learning models such as CLIP [139] and BLIP2 [95], achieved state-of-the-art performance on image retrieval tasks. These multi-modal models jointly learn vector embeddings for images and text, such that the embedding of the image should be close to that of the text that describes that image. Nonetheless, we have also seen increased interest in obtaining a more symbolic representation of the contents of an image [127, 88]. Datasets like Visual Genome [88] annotate images with *scene graphs*. A scene graph is a small KG that describes, with a structured formal graphical representation, the contents of an image in terms of objects (people, animals, items) as nodes connected via pairwise relationships (e.g., actions or positioning relationships) as edges. Therefore, Multimodal LLM can be trained to reason and exploit this additional representation offering an advanced ability to understand the contents of an image (or a video). Importantly, a scene graph node and edge can be annotated (grounded) with features and positions from the image (e.g., relative size). This can be exploited in applications like image and video search and question answering. When entities in a scene graph are connected to a background taxonomy or KG, then questions that require forms of abstraction and computation, e.g., *What is the genus of the largest animal in the picture? What is the average price of the car in the picture? Retrieve images depicting kids wearing vegan friendly items of clothing in the catalog.*

On the other hand, the digitalization of domain specific documents, e.g., especially contracts, is enabling in-depth applications of machine intelligence to help humans more effectively perform time-consuming tasks. Among these, contract review costs humans substantial time, money, and attention (many law firms spend approximately 50% of their time reviewing contracts, costing hundreds of thousands of dollars) [63]. The Contract Understanding Atticus Dataset (CUAD) is a new dataset for legal contract review [63]. CUAD was created with legal experts and consisted of over 13,000 annotations. Recent advancements in generic pre-trained language models showed their power in some text-understanding tasks [182, 186]. Therefore, we have seen different domain specific applications of NLP and LLM methods [182, 186]. Yet, they do not have access to all relevant knowledge and are ill-suited for certain calculations [81]. This can be solved by exploiting external domain specific symbolic information, e.g., domain specific knowledge graphs [117, 99],

[5] E.g., `https://haystack.deepset.ai/`

and by adding symbolic and reasoning capabilities to the LLMs [81]. This promising direction will lead to extending current state-of-the-art neuro-symbolic methods to combine the advantages of a graph representation when extracting symbolic knowledge from complex documents, the ability to understand long-form unstructured texts of LLMs, and the good properties of domain-specific code synthesis of LLMs to address advanced retrieval and question answering use-cases, e.g., *How many weeks are left before the expiration of this contract? What are the conflicts between this liability clause and previous contracts my company signed? What is the risk of side-effects of this treatment for a patient with this given health record? Get the average yearly yield and spread for competitor companies of ACorp and their recent acquisitions in the EU market.*

5 Outlook

In conclusion, the recent advances on large language models (LLMs) mark an important inflection point for knowledge graph (KG) research. While important questions on the ability to combine their strengths remain open, these offer exciting opportunities for future research. The community is already rapidly adapting their research focus, with novel forums like the KBC-LM workshop [79] and the LM-KBC challenge [151] arising, and resources massively shifting towards hybrid approaches to knowledge extraction, consolidation, and usage. We give out the following recommendations:

1. *Don't throw out the KG with the paradigm shift*: For a range of reliability or safety-critical applications, structured knowledge remains indispensible, and we have outlined many ways in which KGs and LLMs can fertilize each other. KGs are here to stay, do not just ditch them out of fashion.
2. *Murder your (pipeline) darlings*: LLMs have substantially advanced many tasks in the KG and ontology construction pipeline, and even made some tasks obsolete. Take critical care in examining even the most established pipeline components, and compare them continuously with the LLM-based state of the art.
3. *Stay curious, stay critical*: LLMs are arguably the most impressive artifact of AI research of the past years. Nonetheless, there exist a magnitude of exaggerated claims and expectations in the public as well as in the research literature, and one should retain a healthy dose of critical reflection. In particular, a fundamental fix to the so-called problem of hallucinations is not in sight.
4. *The past is over, let's begin the new journey*: The advances triggered by LLMs have uprooted the field in an unprecedented manner, and enable to enter the field with significant shortcuts. There is no better time to start anew in fields related to Knowledge Computing, than now.

Although the direction of the present transformation is widely open, as researchers continue to explore the potentials and challenges of hybrid approaches, we can expect to see new breakthroughs in the representation and processing of knowledge, with far-reaching implications for fields ranging from Knowledge Computing to NLP, AI, and beyond.

References

1 Oshin Agarwal, Heming Ge, Siamak Shakeri, et al. Knowledge graph based synthetic corpus generation for knowledge-enhanced language model pretraining. In *NAACL*, pages 3554–3565, June 2021.

2 Naser Ahmadi, Viet-Phi Huynh, Vamsi Meduri, Stefano Ortona, and Paolo Papotti. Mining expressive rules in knowledge graphs. *J. Data and Information Quality*, 12(2), may 2020.

3 Mirza Mohtashim Alam, Md Rashad Al Hasan Rony, Mojtaba Nayyeri, Karishma Mohiuddin, MST Mahfuja Akter, Sahar Vahdati, and Jens Lehmann. Language model guided knowledge graph embeddings. *IEEE Access*, 10:76008–76020, 2022.

4 Dimitrios Alivanistos, Selene Báez Santamaría, Michael Cochez, Jan-Christoph Kalo, Emile van

Krieken, and Thiviyan Thanapalasingam. Prompting as probing: Using language models for knowledge base construction. *LM-KBC*, 2022.

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

6 Mona Alshahrani, Mohammad Asif Khan, Omar Maddouri, Akira R Kinjo, Núria Queralt-Rosinach, and Robert Hoehndorf. Neuro-symbolic representation learning on biological knowledge graphs. *Bioinformatics*, 33(17):2723–2730, 04 2017.

7 Sihem Amer-Yahia, Angela Bonifati, Lei Chen, Guoliang Li, Kyuseok Shim, Jianliang Xu, and Xiaochun Yang. From large language models to databases and back: A discussion on research and education. *DASFAA*, abs/2306.01388, 2023.

8 Alejandro Barredo Arrieta, Natalia Díaz Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, A. Barbado, Salvador García, Sergio Gil-Lopez, Daniel Molina, Richard Benjamins, Raja Chatila, and Francisco Herrera. Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, 2020.

9 Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735, 2007.

10 Stephen H Bach, Victor Sanh, Zheng-Xin Yong, Albert Webson, Colin Raffel, Nihal V Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Fevry, et al. Promptsource: An integrated development environment and repository for natural language prompts. *ACL*, 2022.

11 Parishad BehnamGhader, Santiago Miret, and Siva Reddy. Can retriever-augmented language models reason? the blame game between the retriever and the language model. In *arXiv*, 2022.

12 Emily M Bender, Timnit Gebru, Angelina McMillan-Major, et al. On the dangers of stochastic parrots: Can language models be too big? In *FAT*, pages 610–623, 2021.

13 Emily M. Bender and Alexander Koller. Climbing towards NLU: On meaning, form, and understanding in the age of data. In *ACL*, 2020.

14 Russa Biswas, Harald Sack, and Mehwish Alam. Madlink: Attentive multihop and entity descriptions for link prediction in knowledge graphs. *SWJ*, pages 1–24, 2022.

15 Russa Biswas, Radina Sofronova, Mehwish Alam, and Harald Sack. Contextual language models for knowledge graph completion. In *MLSMKG*, 2021.

16 Su Lin Blodgett, Solon Barocas, Hal Daum'e, and Hanna M. Wallach. Language (technology) is power: A critical survey of "bias" in nlp. *ACL*, 2020.

17 Bernd Bohnet, Vinh Q Tran, Pat Verga, Roee Aharoni, Daniel Andor, Livio Baldini Soares, Jacob Eisenstein, Kuzman Ganchev, Jonathan Herzig, Kai Hui, et al. Attributed question answering: Evaluation and modeling for attributed large language models. *arXiv preprint arXiv:2212.08037*, 2022.

18 Angela Bonifati, Wim Martens, and Thomas Timm. Navigating the maze of wikidata query logs. In *WWW*, pages 127–138, 2019.

19 Angela Bonifati, Wim Martens, and Thomas Timm. An analytical study of large SPARQL query logs. *VLDB J.*, 29(2-3):655–679, 2020.

20 Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. COMET: Commonsense transformers for automatic knowledge graph construction. In *ACL*, pages 4762–4779, 2019.

21 Ruben Branco, António Branco, João António Rodrigues, et al. Shortcutted commonsense: Data spuriousness in deep learning of commonsense reasoning. In *EMNLP*, pages 1504–1521, November 2021.

22 Ryan Brate, Minh-Hoang Dang, Fabian Hoppe, Yuan He, Albert Meroño-Peñuela, and Vijay Sadashivaiah. Improving language model predictions via prompts enriched with knowledge graphs. In *DL4KG*, 2022.

23 Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *neurIPS*, 33:1877–1901, 2020.

24 Boxi Cao, Hongyu Lin, Xianpei Han, Le Sun, Lingyong Yan, Meng Liao, Tong Xue, and Jin Xu. Knowledgeable or educated guess? revisiting language models as knowledge bases. In *ACL*, pages 1860–1874, August 2021.

25 J Harry Caufield, Harshad Hegde, Vincent Emonet, Nomi L Harris, Marcin P Joachimiak, Nicolas Matentzoglu, HyeongSik Kim, Sierra AT Moxon, Justin T Reese, Melissa A Haendel, et al. Structured prompt interrogation and recursive extraction of semantics (spires): A method for populating knowledge bases using zero-shot learning. *arXiv*, 2023.

26 Jiaoyan Chen, Yuan He, Yuxia Geng, Ernesto Jiménez-Ruiz, Hang Dong, and Ian Horrocks. Contextual semantic embeddings for ontology subsumption prediction. *WWW*, pages 1–23, 2023.

27 Jiaoyan Chen, Ernesto Jiménez-Ruiz, Ian Horrocks, Denvar Antonyrajah, Ali Hadian, and Jaehun Lee. Augmenting ontology alignment by semantic embedding and distant supervision. In *ESWC*, pages 392–408, 2021.

28 Jiaoyan Chen, Freddy Lecue, Jeff Z. Pan, Ian Horrocks, and Huajun Chen. Knowledge-based Transfer Learning Explanation. In *KR*, pages 349–358, 2018.

29 Jiaqiang Chen, Niket Tandon, Charles Darwis Hariman, and Gerard de Melo. WebBrain: Joint neural learning of large-scale commonsense knowledge. In *ISWC*, pages 102–118, 2016.

30 Wenhu Chen, Yu Su, Xifeng Yan, and William Yang Wang. Kgpt: Knowledge-grounded pretraining for data-to-text generation. In *EMNLP*, 2020.

31 Xiang Chen, Ningyu Zhang, Xin Xie, Shumin Deng, Yunzhi Yao, Chuanqi Tan, Fei Huang, Luo Si, and Huajun Chen. Knowprompt: Knowledge-aware prompt-tuning with synergistic optimization for relation extraction. In *WWW*, pages 2778–2788, 2022.

32 Bongggeun Choi, Daesik Jang, and Youngjoong Ko. Mem-kgc: Masked entity model for knowledge graph completion with pre-trained language model. *IEEE Access*, 9, 2021.

33 Bonggeun Choi and Youngjoong Ko. Knowledge graph extension with a pre-trained language model via unified learning method. *Knowledge-Based Systems*, page 110245, 2023.

34 Nurendra Choudhary and Chandan K Reddy. Complex logical reasoning over knowledge graphs using large language models. *arXiv*, 2023.

35 Aida Mostafazadeh Davani, Mark Díaz, and Vinodkumar Prabhakaran. Dealing with disagreements: Looking beyond the majority vote in subjective annotations. *TACL*, 10:92–110, 2022.

36 Daniel Daza, Michael Cochez, and Paul Groth. Inductive Entity Representations from Text via Link Prediction. In *WWW*, pages 798–808, 2021.

37 Nicola De Cao, Wilker Aziz, and Ivan Titov. Editing factual knowledge in language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6491–6506, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

38 Xiang Deng, Huan Sun, Alyssa Lees, You Wu, and Cong Yu. Turl: Table understanding through representation learning. *ACM SIGMOD Record*, 51(1):33–40, 2022.

39 Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *NAACL*, 2019.

40 Xin Luna Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Kevin Murphy, Shaohua Sun, and Wei Zhang. From data fusion to knowledge fusion. *VLDB*, 7(10):881–892, jun 2014.

41 Cícero Nogueira dos Santos, Zhe Dong, Daniel Matthew Cer, John Nham, Siamak Shakeri, Jianmo Ni, and Yun-Hsuan Sung. Knowledge prompts: Injecting world knowledge into language models through soft prompts. *ArXiv*, 2022.

42 Mengnan Du, Fengxiang He, Na Zou, et al. Shortcut learning of large language models in natural language understanding: A survey. *arXiv*, 2022.

43 Yupei Du, Qi Zheng, Yuanbin Wu, Man Lan, Yan Yang, and Meirong Ma. Understanding gender bias in knowledge base embeddings. In *ACL*, 2022.

44 Yanai Elazar, Nora Kassner, Shauli Ravfogel, Amir Feder, Abhilasha Ravichander, Marius Mosbach, Yonatan Belinkov, Hinrich Schütze, and Yoav Goldberg. Measuring causal effects of data statistics on language model's 'factual' predictions. *arXiv*, 2023.

45 Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. Measuring and improving consistency in pretrained language models. *TACL*, 9, 2021.

46 Ahmed K. Elmagarmid, Panagiotis G. Ipeirotis, and Vassilios S. Verykios. Duplicate record detection: A survey. *TKDE*, 19(1):1–16, 2007.

47 Grace Fan, Jin Wang, Yuliang Li, Dan Zhang, and Renée Miller. Semantics-aware dataset discovery from data lakes with contextualized column-based representation learning. *VLDB*, 2023.

48 Wenfei Fan, Chunming Hu, Xueli Liu, and Ping Lu. Discovering graph functional dependencies. *TODS*, 45(3), 2020.

49 Wenfei Fan, Ping Lu, Chao Tian, and Jingren Zhou. Deducing certain fixes to graphs. *VDLB*, 12(7):752–765, 2019.

50 I. P. Fellegi and A. B. Sunter. A theory for record linkage. *Journal of the American Statistical Association*, 64:1183–1210, 1969.

51 Besnik Fetahu, Ujwal Gadiraju, and Stefan Dietze. Improving entity retrieval on structured data. In *ISWC*, pages 474–491, 2015.

52 Luis Antonio Galárraga, Christina Teflioudi, Katja Hose, and Fabian Suchanek. Amie: Association rule mining under incomplete evidence in ontological knowledge bases. In *WWW*, WWW '13, page 413–422, 2013.

53 Daniel Gao, Yantao Jia, Lei Li, Chengzhen Fu, Zhicheng Dou, Hao Jiang, Xinyu Zhang, Lei Chen, and Zhao Cao. Kmir: A benchmark for evaluating knowledge memorization, identification and reasoning abilities of language models, 2022.

54 Genet Asefa Gesese, Russa Biswas, Mehwish Alam, and Harald Sack. A survey on knowledge graph embeddings with literals: Which model links better literal-ly? *Semantic Web*, 12(4):617–647, 2021.

55 Bernardo Cuenca Grau, Ian Horrocks, Boris Motik, Bijan Parsia, Peter F. Patel-Schneider, and Ulrike Sattler. OWL 2: The next step for OWL. *J. Web Semant*, 6(4):309–322, 2008.

56 Paul Groth, Elena Paslaru Bontas Simperl, Marieke van Erp, and Denny Vrandecic. Knowledge graphs and their role in the knowledge engineering of the 21st century (dagstuhl seminar 22372). *Dagstuhl Reports*, 12:60–120, 2022.

57 Xiaoqi Han, Ru Li, Xiaoli Li, and Jeff Z. Pan. A Divide and Conquer Framework for Knowledge Editing. *Knowledge Based Systems*, 2023.

58 Yuan He, Jiaoyan Chen, Denvar Antonyrajah, and Ian Horrocks. Bertmap: a bert-based ontology alignment system. In *AAAI*, volume 36, pages 5684–5691, 2022.

59 Yuan He, Jiaoyan Chen, Hang Dong, Ian Horrocks, Carlo Allocca, Taehun Kim, and Brahmananda Sapkota. DeepOnto: A Python package for ontology engineering with deep learning. *arXiv preprint arXiv:2307.03067*, 2023.

60 Yuan He, Jiaoyan Chen, Hang Dong, Ernesto Jiménez-Ruiz, Ali Hadian, and Ian Horrocks. Machine learning-friendly biomedical datasets for equivalence and subsumption ontology matching. In *ISWC*, pages 575–591, 2022.

61 Yuan He, Jiaoyan Chen, Ernesto Jiménez-Ruiz, Hang Dong, and Ian Horrocks. Language model analysis for ontology subsumption inference. In *Findings of ACL*, 2023.

62 Benjamin Heinzerling and Kentaro Inui. Language models as knowledge bases: On entity representations, storage capacity, and paraphrased queries. In *EACL*, pages 1772–1791, 2021.

63 Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. Cuad: An expert-annotated nlp dataset for legal contract review. In *neurIPS*, volume 1, 2021.

64 Or Honovich, Uri Shaham, Samuel R Bowman, et al. Instruction induction: From few examples to natural language task descriptions. *arXiv*, 2022.

65 Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *ICLR*, 2021.

66 Ziniu Hu, Yichong Xu, Wenhao Yu, Shuohang Wang, Ziyi Yang, Chenguang Zhu, Kai-Wei Chang, and Yizhou Sun. Empowering language models with knowledge graph reasoning for question answering. In *New Frontiers in Graph Learning*, 2022.

67 Jian Huang, Jianfeng Gao, Jiangbo Miao, Xiaolong Li, Kuansan Wang, Fritz Behr, and C. Lee Giles. Exploring web scale language models for search query processing. In *WWW*, page 451–460, 2010.

68 Jie Huang and Kevin Chen-Chuan Chang. Towards reasoning in large language models: A survey. *Findings of ACL*, 2023.

69 Ningyuan Huang, Yash R Deshpande, Yibo Liu, Houda Alberts, Kyunghyun Cho, Clara Vania, and Iacer Calixto. Endowing language models with multimodal knowledge graph representations. *arXiv*, 2022.

70 Jena D. Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. Comet-atomic 2020: On symbolic and neural commonsense knowledge graphs. In *AAAI*, 2021.

71 Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. In *EACL*, 2021.

72 Sarthak Jain and Byron C. Wallace. Attention is not explanation. In *NAACL*, 2019.

73 Krzysztof Janowicz, Bo Yan, Blake Regalia, Rui Zhu, and Gengchen Mai. Debiasing knowledge graphs: Why female presidents are not like female popes. In *International Workshop on the Semantic Web*, 2018.

74 Zhengbao Jiang, Frank F Xu, Jun Araki, et al. How can we know what language models know? *TACL*, 8:423–438, 2020.

75 Ernesto Jiménez-Ruiz and Bernardo Cuenca Grau. Logmap: Logic-based and scalable ontology matching. In *ISWC*, pages 273–288, 2011.

76 Ernesto Jiménez-Ruiz, Oktie Hassanzadeh, Vasilis Efthymiou, Jiaoyan Chen, and Kavitha Srinivas. Semtab 2019: Resources to benchmark tabular data to knowledge graph matching systems. In *ESWC*, pages 514–530, 2020.

77 Martin Josifoski, Marija Sakota, Maxime Peyrard, and Robert West. Exploiting asymmetry for synthetic training data generation: Synthie and the case of information extraction. *ArXiv*, 2023.

78 Jan-Christoph Kalo and Leandra Fichtel. Kamel: Knowledge analysis with multitoken entities in language models. In *AKBC*, 2022.

79 Jan-Christoph Kalo, Simon Razniewski, Sneha Singhania, and Jeff Z. Pan. LM-KBC: Knowledge base construction from pre-trained language models. *ISWC Challenges*, 2023.

80 Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. Large language models struggle to learn long-tail knowledge, 2023.

81 Ehud Karpas, Omri Abend, Yonatan Belinkov, Barak Lenz, Opher Lieber, Nir Ratner, Yoav Shoham, Hofit Bata, Yoav Levine, Kevin Leyton-Brown, et al. Mrkl systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning. *arXiv*, 2022.

82 Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen tau Yih. Dense passage retrieval for open-domain question answering. In *EMNLP*, page 6769–6781, 2020.

83 Nora Kassner and Hinrich Schütze. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *ACL*, 2020.

84 Bosung Kim, Taesuk Hong, Youngjoong Ko, and Jungyun Seo. Multi-task learning for knowledge graph completion with pre-trained language models. In *COLING*, pages 1737–1743, 2020.

85 Holger Knublauch and Dimitris Kontokostas. Shapes constraint language (shacl). *W3C Candidate Recommendation*, 11(8), 2017.

86 Keti Korini and Christian Bizer. Column type annotation using chatgpt. *arXiv*, 2023.

87 Angelie Kraft and Ricardo Usbeck. The lifecycle of "facts": A survey of social bias in knowledge graphs. In *AACL*, 2022.

88 Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 123:32–73, 2017.

89 Viet Dac Lai, Nghia Trung Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Huu Nguyen. Chatgpt beyond english: Towards a comprehensive evaluation of large language models in multilingual learning. *ArXiv*, 2023.

90 Yoonjoo Lee, John Joon Young Chung, Tae Soo Kim, Jean Y. Song, and Juho Kim. Promptiverse: Scalable generation of scaffolding prompts through human-ai hybrid knowledge graph annotation. In *CHI*, 2022.

91 Alina Leidinger and Richard A. Rogers. Which stereotypes are moderated and under-moderated in search engine autocompletion? *FAT*, 2023.

92 Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *neurIPS*, volume 33, 2020.

93 Bo Li, Gexiang Fang, Yang Yang, Quansen Wang, Wei Ye, Wen Zhao, and Shikun Zhang. Evaluating chatgpt's information extraction capabilities: An assessment of performance, explainability, calibration, and faithfulness. *ArXiv*, 2023.

94 Da Li, Ming Yi, and Yukai He. Lp-bert: Multi-task pre-training knowledge graph bert for link prediction. *arXiv*, 2022.

95 Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *arXiv*, 2023.

96 Tianyi Li, Mohammad Javad Hosseini, Sabine Weber, and Mark Steedman. Language models are poor learners of directional inference. In *Findings in EMNLP*, 2022.

97 Tianyi Li, Wenyu Huang, Nikos Papasarantopoulos, Pavlos Vougiouklis, and Jeff Z. Pan. askspecific pre-training and prompt decomposition for knowledge graph population with language models. In *LM-KBC*, 2022.

98 Stephan Linzbach, Tim Tressel, Laura Kallmeyer, Stefan Dietze, and Hajira Jabeen. Decoding prompt syntax: Analysing its impact on knowledge retrieval in large language models. In *NLP4KGC*, 2023.

99 Matteo Lissandrini, Davide Mottin, Themis Palpanas, Dimitra Papadimitriou, and Yannis Velegrakis. Unleashing the power of information graphs. *SIGMOD Record*, 43(4):21–26, 2015.

100 Fangyu Liu, Ivan Vulic, Anna Korhonen, and Nigel Collier. Learning domain-specialised representations for cross-lingual biomedical entity linking. In *ACL*, 2021.

101 Hao Liu, Yehoshua Perl, and James Geller. Concept placement using bert trained by transforming and summarizing biomedical ontology structure. *Journal of Biomedical Informatics*, 112:103607, 2020.

102 Hugo Liu and Push Singh. Commonsense reasoning in and over natural language. In *Knowledge-Based Intelligent Information and Engineering Systems*, pages 293–306, 2004.

103 Jixiong Liu, Yoan Chabot, Raphaël Troncy, Viet-Phi Huynh, Thomas Labbé, and Pierre Monnin. From tabular data to knowledge graphs: A survey of semantic table interpretation tasks and methods. *JWS*, 2022.

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

105 Fengyuan Lu, Peijin Cong, and Xinli Huang. Utilizing textual information in knowledge graph embedding: A survey of methods and applications. *IEEE Access*, 8:92072–92088, 2020.

106 Chaitanya Malaviya, Chandra Bhagavatula, Antoine Bosselut, and Yejin Choi. Commonsense knowledge base completion with structural and semantic context. In *AAAI*, volume 34, pages 2925–2933, 2020.

107 Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *ACL*, 2023.

108 Rui Mao, Qian Liu, Kai He, et al. The biases of pre-trained language models: An empirical study on prompt-based sentiment analysis and emotion detection. *IEEE Transactions on Affective Computing*, pages 1–11, 2022.

109 Ninareh Mehrabi, Thamme Gowda, Fred Morstatter, Nanyun Peng, and A. G. Galstyan. Man is to person as woman is to location: Measuring gender bias in named entity recognition. *Conference on Hypertext and Social Media*, 2019.

110 Chris Mellish and Jeff Z. Pan. Natural Language Directed Inference from Ontologie". In *Artificial Intelligence Journal*, 2008.

111 Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D Manning, and Chelsea Finn. Memory-based model editing at scale. In *International Conference on Machine Learning*, pages 15817–15831. PMLR, 2022.

112 Fedor Moiseev, Zhe Dong, Enrique Alfonseca, and Martin Jaggi. Skill: Structured knowledge infusion for large language models. In *NAACL*, 2022.

113 Moin Nadeem, Anna Bethke, and Siva Reddy. Stereoset: Measuring stereotypical bias in pretrained language models. In *ACL*, 2020.

114 Mojtaba Nayyeri, Zihao Wang, Mst Akter, Mirza Mohtashim Alam, Md Rashad Al Hasan Rony, Jens Lehmann, Steffen Staab, et al. Integrating knowledge graph embedding and pretrained language models in hypercomplex spaces. *arXiv*, 2022.

115 Sophie Neutel and Maaike HT de Boer. Towards automatic ontology alignment using bert. In *AAAI Spring Symposium: Combining Machine Learning with Knowledge Engineering*, 2021.

116 Tuan-Phong Nguyen, Simon Razniewski, Aparna Varde, and Gerhard Weikum. Extracting cultural commonsense knowledge at scale. In *WWW*, page 1907–1917, 2023.

117 Natasha Noy, Yuqing Gao, Anshu Jain, Anant Narayanan, Alan Patterson, and Jamie Taylor. Industry-scale knowledge graphs: lessons and challenges. *Communications of the ACM*, 62(8):36–43, 2019.

118 OpenAI. Gpt-4 technical report, 2023.

119 Long Ouyang, Jeff Wu, Xu Jiang, et al. Training language models to follow instructions with human feedback. *neurIPS*, 2022.

120 Barlas Oğuz, Xilun Chen, Vladimir Karpukhin, Stanislav Peshterliev, Dmytro Okhonko, M. Schlichtkrull, Sonal Gupta, Yashar Mehdad, and Scott Yih. Unik-qa: Unified representations of structured and unstructured knowledge for open-domain question answering. In *NAACL-HLT*, 2020.

121 Jeff Z. Pan. Resource Description Framework. In *Handbook on Ontologies*. IOS Press, 2009.

122 Jeff Z. Pan and Ian Horrocks. Web Ontology Reasoning with Datatype Groups. In *ISWC*, pages 47–63, 2003.

123 Jeff Z. Pan, Guido Vetere, José Manuél Gómez-Pérez, and Honghan Wu. Exploiting linked data and knowledge graphs in large organisations. *Springer International Publishing*, 2017.

124 Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. Unifying large language models and knowledge graphs: A roadmap. *arXiv*, 2023.

125 Lalchand Pandia and Allyson Ettinger. Sorting through the noise: Testing robustness of information processing in pre-trained language models. *EMNLP*, 2021.

126 George Papadakis, Ekaterini Ioannou, Emanouil Thanos, and Themis Palpanas. The four generations of entity resolution. In *Synthesis Lectures on Data Management*, 2021.

127 Jae Sung Park, Chandra Bhagavatula, Roozbeh Mottaghi, Ali Farhadi, and Yejin Choi. Visualcomet: Reasoning about the dynamic context of a still image. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *ECCV*, pages 508–524, 2020.

128 Heiko Paulheim. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic Web*, 8(3):489–508, 2017.

129 Shichao Pei, Lu Yu, Guoxian Yu, and Xiangliang Zhang. Rea: Robust cross-lingual entity alignment between knowledge graphs. In *KDD*, page 2175–2184, 2020.

130 Matthew E Peters, Mark Neumann, Robert L Logan IV, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A Smith. Knowledge enhanced contextual word representations. *EMNLP*, 2019.

131 Fabio Petroni, Patrick Lewis, Aleksandra Piktus, et al. How context affects language models' factual predictions. *AKBC*, 2020.

132 Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, et al. Language models as knowledge bases? In *EMNLP*, pages 2463–2473, 2019.

133 Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language Models as Knowledge Bases? In *EMNLP*, pages 2463–2473, 2019.

134 Barbara Plank. The "problem" of human label variation: On ground truth in data, modeling and evaluation. *EMNLP*, abs/2211.02570, 2022.

135 Eric Prud'hommeaux, José Emilio Labra Gayo, and Harold R. Solbrig. Shape expressions: an RDF validation and transformation language. In *SEMANTiCS*, pages 32–40, 2014.

136 Kashif Rabbani, Matteo Lissandrini, and Katja Hose. Shacl and shex in the wild: A community survey on validating shapes generation and adoption. In *WWW*, pages 260–263, 2022.

137 Kashif Rabbani, Matteo Lissandrini, and Katja Hose. Extraction of validating shapes from very large knowledge graphs. *VLDB*, 16(5):1023–1032, 2023.

138 Kashif Rabbani, Matteo Lissandrini, and Katja Hose. Shactor: Improving the quality of large-scale knowledge graphs with validating shapes. *Companion of the 2023 International Conference on Management of Data*, 2023.

139 Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, volume 139 of *PMLR*, pages 8748–8763, 2021.

140 Leonardo Ranaldi, Elena Sofia Ruzzetti, David A. Venditti, Dario Onorati, and Fabio Massimo Zanzotto. A trip towards fairness: Bias and de-biasing in large language models. *ArXiv*, 2023.

141 Abhilasha Ravichander, Eduard Hovy, Kaheer Suleman, Adam Trischler, and Jackie Chi Kit Cheung. On the systematicity of probing contextualized word representations: The case of hypernymy in bert. In *Joint Conference on Lexical and Computational Semantics*, pages 88–102, 2020.

142 Leonardo F. R. Ribeiro, Martin Schmitt, Hinrich Schütze, and Iryna Gurevych. Investigating pretrained language models for graph-to-text generation. *Workshop on Natural Language Processing for Conversational AI*, abs/2007.08426, 2021.

143 Devendra Sachan, Yuhao Zhang, Peng Qi, et al. Do syntax trees help pre-trained transformers extract information? In *EACL*, pages 2647–2661, April 2021.

144 Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. The risk of racial bias in hate speech detection. In *ACL*, 2019.

145 Patrick Schramowski, Cigdem Turan, Nico Andersen, et al. Large pre-trained language models contain human-like biases of what is right and wrong to do. *Nature Machine Intelligence*, 4(3):258–268, 2022.

146 Jingyu Shao, Qing Wang, Asiri Wijesinghe, and Erhard Rahm. Ergan: Generative adversarial networks for entity resolution. In *ICDM*, pages 1250–1255, 2020.

147 Jingchuan Shi, Jiaoyan Chen, Hang Dong, Ishita Khan, Lizzie Liang, Qunzhi Zhou, Zhe Wu, and Ian Horrocks. Subsumption prediction for e-commerce taxonomies. In *ESWC*, pages 244–261, 2023.

148 Peng Shi and Jimmy J. Lin. Simple bert models for relation extraction and semantic role labeling. *ArXiv*, abs/1904.05255, 2019.

149 Jaeho Shin, Sen Wu, Feiran Wang, Christopher De Sa, Ce Zhang, and Christopher Ré. Incremental knowledge base construction using deepdive. In *VLDB*, volume 8, page 1310, 2015.

150 Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *EMNLP*, pages 4222–4235, 2020.

151 Sneha Singhania, Tuan-Phong Nguyen, and Simon Razniewski. LM-KBC: Knowledge base construction from pre-trained language models. *Semantic Web challenge*, 2022.

152 Sneha Singhania, Simon Razniewski, and Gerhard Weikum. Predicting Document Coverage for Relation Extraction. *Transactions of the Association for Computational Linguistics*, 10:207–223, 03 2022.

153 Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *TMLR*, 2023.

154 Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. Yago: a core of semantic knowledge. In *The Web Conference*, 2007.

155 Yoshihiko Suhara, Jinfeng Li, Yuliang Li, Dan Zhang, Çağatay Demiralp, Chen Chen, and Wang-Chiew Tan. Annotating columns with pre-trained language models. In *SIGMOD*, pages 1493–1503, 2022.

156 Zequn Sun, Qingheng Zhang, Wei Hu, Chengming Wang, Muhao Chen, Farahnaz Akrami, and Chengkai Li. A benchmarking study of embedding-based entity alignment for knowledge graphs. *VLDB*, 13(11):2326–2340, 2020.

157 Alexandre Tamborrino, Nicola Pellicano, Baptiste Pannier, et al. Pre-training is (almost) all you need: An application to commonsense reasoning. *ACL*, 2020.

158 Wang-Chiew Tan, Yuliang Li, Pedro Rodriguez, Richard James, Xi Victoria Lin, Alon Halevy, and Scott Yih. Reimagining retrieval augmented language models for answering queries. In *Findings of ACL*, page 6131–6146, 2023.

159 Niket Tandon and Gerard de Melo. Information extraction from web-scale n-gram data. In *Web N-gram Workshop*, volume 5803, pages 8–15, 2010.

160 Niket Tandon, Gerard de Melo, Fabian M. Suchanek, and Gerhard Weikum. Webchild: harvesting and organizing commonsense knowledge from the web. *Proceedings of the 7th ACM international conference on Web search and data mining*, 2014.

161 Niket Tandon, Gerard de Melo, and Gerhard Weikum. Deriving a Web-scale common sense fact database. In *AAAI*, pages 152–157, 2011.

162 Nan Tang, Ju Fan, Fangyi Li, Jianhong Tu, Xiaoyong Du, Guoliang Li, Sam Madden, and Mourad Ouzzani. Rpt: relational pre-trained transformer is almost all you need towards democratizing data preparation. *VLDB*, 2021.

163 Ruixiang Tang, Xiaotian Han, Xiaoqian Jiang, and Xia Hu. Does synthetic data generation of llms help clinical text mining? *ArXiv*, 2023.

164 Nicolas Tempelmeier, Elena Demidova, and Stefan Dietze. Inferring missing categorical information in noisy and sparse web markup. In *WWW*, 2018.

165 Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv*, 2023.

166 Pat Verga, Haitian Sun, Livio Baldini Soares, et al. Facts as experts: Adaptable and interpretable neural memory over symbolic knowledge. *NAACL*, 2021.

167 Blerta Veseli, Sneha Singhania, Simon Razniewski, and Gerhard Weikum. Evaluating language models for knowledge base completion. In *ESWC*, page 227–243, 2023.

168 Liane Vogel, Benjamin Hilprecht, and Carsten Binnig. Towards foundation models for relational databases [vision paper]. *TRL@NeurIPS2022*, 2023.

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

170 David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. Entity, relation, and event extraction with contextualized span representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5784–5789, Hong Kong, China, November 2019. Association for Computational Linguistics.

171 Bin Wang, Guangtao Wang, Jing Huang, Jiaxuan You, Jure Leskovec, and C.-C. Jay Kuo. Inductive learning on commonsense knowledge graph completion. In *Joint Conference on Neural Networks*, pages 1–8, 2021.

172 Bo Wang, Tao Shen, Guodong Long, Tianyi Zhou, Ying Wang, and Yi Chang. Structure-augmented text representation learning for efficient knowledge graph completion. In *WWW*, pages 1737–1748, 2021.

173 Liang Wang, Wei Zhao, Zhuoyu Wei, and Jingming Liu. SimKGC: Simple Contrastive Knowledge Graph Completion with Pre-trained Language Models. In *ACL*, pages 4281–4294, May 2022.

174 Xiao Wang, Wei Zhou, Can Zu, Han Xia, Tianze Chen, Yuan Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, Jihua Kang, J. Yang, Siyuan Li, and Chunsai Du. Instructuie: Multi-task instruction tuning for unified information extraction. *ArXiv*, 2023.

175 Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. KEPLER: A Unified Model for Knowledge Embedding and Pre-trained Language Representation. *TACL*, 9:176–194, March 2021.

176 Xiyu Wang and Nora El-Gohary. Deep learning-based relation extraction and knowledge graph-based representation of construction safety requirements. *Automation in Construction*, 147:104696, 2023.

177 Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Huai hsin Chi, F. Xia, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903, 2022.

178 Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, Yong Jiang, and Wenjuan Han. Zero-shot information extraction via chatting with chatgpt. *ArXiv*, 2023.

179 Gerhard Weikum, Luna Dong, Simon Razniewski, and Fabian M. Suchanek. Machine Knowledge: Creation and Curation of Comprehensive Knowledge Bases. In *FnT*, 2021.

180 Joseph Weizenbaum. Eliza—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 1966.

181 Peter West, Chandra Bhagavatula, Jack Hessel, Jena D Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. Symbolic knowledge distillation: from general language models to commonsense models. *arXiv preprint arXiv:2110.07178*, 2021.

182 Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. Bloomberggpt: A large language model for finance. *arXiv*, 2023.

183 Guohui Xiao, Diego Calvanese, Roman Kontchakov, Domenico Lembo, Antonella Poggi, Riccardo Rosati, and Michael Zakharyaschev. Ontology-based data access: A survey. In *IJCAI*, pages 5511–5519, 2018.

184 Yang Xu, Mahdi Namazifar, Devamanyu Hazarika, Aishwarya Padmakumar, Yang Liu, and Dilek Z.

Hakkani-Tür. Kilm: Knowledge injection into encoder-decoder language models. *ArXiv*, 2023.

185 Linyao Yang, Hongyang Chen, Zhao Li, Xiao Ding, and Xindong Wu. Chatgpt is not enough: Enhancing large language models with knowledge graphs for fact-aware language modeling. *arXiv*, 2023.

186 Xi Yang, Aokun Chen, Nima PourNejatian, Hoo Chang Shin, Kaleb E. Smith, Christopher Parisien, Colin Compas, Cheryl Martin, Anthony B. Costa, Mona G. Flores, Ying Zhang, Tanja Magoc, Christopher A. Harle, Gloria Lipori, Duane A. Mitchell, William R. Hogan, Elizabeth A. Shenkman, Jiang Bian, and Yonghui Wu. A large language model for electronic health records. *npj Digital Medicine*, 5(1):194, 2022.

187 Liang Yao, Chengsheng Mao, and Yuan Luo. Kg-bert: Bert for knowledge graph completion. *arXiv*, 2019.

188 Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. Qa-gnn: Reasoning with language models and knowledge graphs for question answering. *NAACL*, 2021.

189 Donghan Yu, Chenguang Zhu, Yuwei Fang, Wenhao Yu, Shuohang Wang, Yichong Xu, Xiang Ren, Yiming Yang, and Michael Zeng. Kg-fid: Infusing knowledge graph in fusion-in-decoder for open-domain question answering. In *ACL*, 2022.

190 Ran Yu, Ujwal Gadiraju, Besnik Fetahu, Oliver Lehmberg, Dominique Ritze, and Stefan Dietze. Knowmore - knowledge base augmentation with structured web markup. *Semantic Web*, 10(1):159–180, 2019.

191 Qingkai Zeng, Jinfeng Lin, Wenhao Yu, Jane Cleland-Huang, and Meng Jiang. Enhancing taxonomy completion with concept generation via fusing relational representations. In *KDD*, pages 2104–2113, 2021.

192 Hanwen Zha, Zhiyu Chen, and Xifeng Yan. Inductive relation prediction by bert. In *AAAI*, volume 36, pages 5923–5931, 2022.

193 Chaoning Zhang, Chenshuang Zhang, Sheng Zheng, Yu Qiao, Chenghao Li, Mengchun Zhang, Sumit Kumar Dam, Chu Myaet Thwal, Ye Lin Tun, Le Luang Huy, Donguk kim, Sung-Ho Bae, Lik-Hang Lee, Yang Yang, Heng Tao Shen, In-So Kweon, and Choong-Seon Hong. A complete survey on generative ai (aigc): Is chatgpt from gpt-4 to gpt-5 all you need? *ArXiv*, 2023.

194 Meiru Zhang, Yixuan Su, Zaiqiao Meng, Zihao Fu, and Nigel Collier. Coffee: A contrastive oracle-free framework for event extraction. *ArXiv*, abs/2303.14452, 2023.

195 Ruichuan Zhang and Nora El-Gohary. Transformer-based approach for automated context-aware ifc-regulation semantic information alignment. *Automation in Construction*, 145, 2023.

196 Wen Zhang, Jiaoyan Chen, Juan Li, Zezhong Xu, Jeff Z Pan, and Huajun Chen. Knowledge graph reasoning with logics and embeddings: survey and perspective. *arXiv*, 2022.

197 Zhiyuan Zhang, Xiaoqian Liu, Yi Zhang, Qi Su, Xu Sun, and Bin He. Pretrain-kge: Learning knowledge representation from pretrained language models. In *EMNLP Findings*, 2020.

198 Ziheng Zhang, Hualuo Liu, Jiaoyan Chen, Xi Chen, Bo Liu, Yuejia Xiang, and Yefeng Zheng. An industry evaluation of embedding-based entity alignment. In *COLING*, pages 179–189, 2020.

199 Zexuan Zhong and Danqi Chen. A frustratingly easy approach for entity and relation extraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 50–61, Online, June 2021. Association for Computational Linguistics.

200 Zexuan Zhong, Dan Friedman, and Danqi Chen. Factual probing is [mask]: Learning vs. learning to recall. In *NAACL*, 2021.

201 Wenxuan Zhou, Fangyu Liu, Ivan Vulic, Nigel Collier, and Muhao Chen. Prix-lm: Pretraining for multilingual knowledge base construction. In *ACL*, 2021.

202 Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. Large language models are human-level prompt engineers. In *arXiv*, 2023.

203 Qi Zhu, Hao Wei, Bunyamin Sisman, Da Zheng, Christos Faloutsos, Xin Luna Dong, and Jiawei Han. Collective multi-type entity alignment between knowledge graphs. In *WWW*, page 2241–2252, 2020.

204 Yuqi Zhu, Xiaohan Wang, Jing Chen, Shuofei Qiao, Yixin Ou, Yunzhi Yao, Shumin Deng, Huajun Chen, and Ningyu Zhang. Llms for knowledge graph construction and reasoning: Recent capabilities and future opportunities. *arXiv*, 2023.