# A Survey of Knowledge Enhanced Pre-trained Models

Jian Yang, Gang Xiao, Yulong Shen, Wei Jiang, Xinyu Hu, Ying Zhang, Jinghui Peng

**Abstract**—Pre-trained models learn contextualized word representations on large-scale text corpus through a self-supervised learning method, which has achieved promising performance after fine-tuning. These models, however, suffer from poor robustness and lack of interpretability. Pre-trained models with knowledge injection, which we call knowledge enhanced pre-trained models (KEPTMs), possess deep understanding and logical reasoning and introduce interpretability to some extent. In this survey, we provide a comprehensive overview of KEPTMs for natural language processing. We first introduce the progress of pre-trained models and knowledge representation learning. Then we systematically categorize existing KEPTMs from three different perspectives. Finally, we outline some potential directions of KEPTMs for future research.

**Index Terms**—Deep learning, knowledge graph, natural language processing, knowledge representation learning, knowledge enhanced pre-trained models.



## 1 Introduction

Data and knowledge are central to artificial intelligence. Deep learning [1], [2], [3] can fully leverage large-scale data by virtue of distributed representation and hierarchical structure generalization of neural networks. Based on deep learning, pre-trained models [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18] have undergone a qualitative leap forward, facilitating a wide range of downstream natural language processing (NLP) applications. Although they can capture lexical, syntactic and shallow semantic information from large-scale unsupervised corpus, they are statistical models and limited by heavy-tailed data distribution, causing the inability of deep understanding and causal and counterfactual reasoning. Besides, pre-trained models lose interpretability because of entangled representations, although deep learning is powerful for learning critical factors behind the data. Knowledge provides comprehensive and rich entity features and relations for models to overcome the influence of the data distribution and enhance its robustness. In addition, knowledge introduces interpretability for models with explicit semantics. Therefore, it is indispensable to utilize different knowledge to achieve pre-trained models with deep understanding and logical reasoning. For better integrating knowledge and text features, symbolic knowledge is projected into a dense, low-dimensional semantic space and presented by distributed vectors thorough knowledge representation learning [19]. In this context, researchers have explored the path towards improving the pre-trained models by injecting knowledge to generalize knowledge-driven and semantic understanding required scenarios.

The contributions of this survey can be summarized as follows:

- *Comprehensive review*. We provide a comprehensive review for pre-trained models and knowledge representation learning for NLP.
- *New taxonomy*. We propose a taxonomy of KEPTMs for NLP, which categorizes existing KEPTMs into three groups according to the type of injected knowledge and further divides corresponding models of different groups according to coupling between knowledge and corpus, and the method of knowledge injection.
- *Future directions*. We discuss and analyze the limitations of existing KEPTMs and suggest some possible future research directions.

The rest of the survey is organized as follows. Section 2 outlines the progress of pre-trained models and knowledge representation learning. Sections 3 introduces a classification basis and a corresponding comprehensive taxonomy. Following Section 3 categorization, Section 4 introduces the working principle of each model in detail and analyzes its pros and cons, and compares existing KEPTMs from different dimensions. Section 5 discusses the current challenges and suggests future directions.

## 2 Background

### 2.1 Pre-trained Models

Deep neural models can be trained through large-scale unsupervised text corpus with skills that we call pre-trained models. The effectiveness of the pre-trained model largely

*(Corresponding authors: Gang Xiao, Yulong Shen)*

- *Jian Yang is with School of Computer Science and Technology, Xidian University, Xi'an 710000, China and with National Key Laboratory for Complex Systems Simulation, Beijing 100000, China. E-mail: seekerferry@gmail.com.*
- *Yulong Shen is with School of Computer Science and Technology, Xidian University, Xi'an 710000, China. E-mail: ylshen@mail.xidian.edu.cn.*
- *Gang Xiao, Wei Jiang, Xinyu Hu, Ying Zhang and Jinghui Peng are with National Key Laboratory for Complex Systems Simulation, Beijing 100000, China. E-mail: searchware@qq.com, loftyjiang@163.com, 1052447409@qq.com, k1814660@kcl.ac.uk, pjh20200828@163.com.*

depends on the representation learning of the model's encoder. Representation learning refers to learning representations of the data that make it easier to extract useful information when building classifiers or other predictors [20]. There are two mainstream paradigms within the community of representation learning: probabilistic graphical models and neural networks. Probabilistic graph models learn feature representation by modeling the posterior distribution of potential variables in sample data, including directed graph model and undirected graph model. Its advantage is that it can model the relationship between some potential variables with prior knowledge, thus bringing interpretability. Neural network models mostly use an autoencoder composed of an encoder and a decoder. The encoder is responsible for feature extraction, while the decoder reconstructs the input by applying a regularized reconstruction objective.

Neural networks based models are preferable with the following advantages than probabilistic graphical models. Firstly, neural networks can express more possible features with distributed vectors instead of sparse vectors. Secondly, considering the existing data is mainly the result of the interaction between multiple latent factors, distributed vectors can represent different impact factors by designing a specific network structure. Finally, the underlying neural layers of deep neural networks transform concrete features learned from data into abstract features in upper layers and keep stable as the local change of the input data, enhancing the robustness of representation to generalize in many downstream tasks.

Following autoencoder-based neural models, pre-trained models design specific neural networks to encode input data while using pre-trained tasks to decode learned representations. After fine-tuning, pre-trained models can easily be adapted to all kinds of NLP tasks. We divide models into token-based models and context-based models according to whether the models capture sequence-level semantic.

#### 2.1.1 Token-based Pre-trained Models

Originating from the NNLM [21] proposed by Bengio in 2003, distributed representations of words are generated as a by-product during the training. According to the hypothesis that words with similar context have similar semantics, Mikolov et al. [4], [5] propose two shallow architectures: Continuous Bag-of-Words (CBOW) and Skip-Gram (SG) to capture latent syntactic and semantic similarities between words. Besides, GloVe [6] computes word-word cooccurrence statistics from a large corpus as a supervised signal, and FastText [7] trains the model with text classification data. With the emergence of all the above token-based pre-trained models, word embeddings have been commonly used as text representation in NLP tasks. Although these models are simple and effective, they are only suited to attain fixed representations rather than capturing polysemy. That is also why we call this type of model static pre-trained models.

#### 2.1.2 Context-based Pre-trained Models

To address the problem of polysemy, pre-trained models need to distinguish the semantics of words and dynamically generate word embeddings in different contexts. Given a text $x_1$,$x_2$,$\cdots$,$x_T$ where each token $x_t$ is a word or subword, the contextual representation of $x_t$ depends on the whole text.

$$[h_1, h_2, \cdots, h_T] = f_{enc}(x_1, x_2, \cdots, x_T), \tag{1}$$

where $f_{enc}(\cdot)$ is neural encoder and $h_t$ is contextual embedding.

Taking LSTM [22] as neural encoder, the ELMo [8] model extracts context-dependent representations from a bidirectional language model, which has shown to bring large improvement on a range of NLP tasks. However, ELMo is usually used as a feature extractor to produce initial embeddings for the main model of downstream tasks, which means the rest parameters of the main model have to train from scratch.

At the same period, the proposal of ULMFiT [23] provides valuable multi-stage transfer and fine-tuned skills for models. Besides, Transformer [24] has achieved surprising success on machine translation and proven to be more effective than LSTM in dealing with long-range text dependencies. In this background, OpenAI proposes GPT [9] that adopts the modified Transformer's decoder as a language model to learn universal representations transferable to a broad range of downstream tasks, which outperforms task-specific architectures in 9 of 12 NLP tasks. GPT-2 and GPT-3 [10], [25] mainly follow the architecture and train on larger and more diverse datasets to learn from varied domains. However, limited by a unidirectional encoder, the GPT series can only attend its left context resulting in sub-optimal for learning sentence-level semantics. To overcome this deficiency, BERT [11] adopts a masked language modelling (MLM) objective where some of the tokens of a sequence are masked randomly, and the goal is to predict these tokens considering the corrupted sentence. Inspired by Skip-Thoughts [26], BERT also employs the next sentence prediction (NSP) task to learn the semantic connection between sentences, which obtains new start-of-art results on eleven NLP tasks and even becomes the basis of subsequent models. Based on BERT, RoBERTa [13] design a few improved training recipes, including training longer with bigger batches over more data, modifying objectives, training over long sequences, and dynamically changing the masking pattern, which enhances significantly performance of BERT. To overcome the discrepancy between pre-training and fine-tuning of BERT, XLNet [12] proposes a new autoregressive method based on permutation language modelling to capture contextual information without introducing any new symbols.

Unlike all these above pre-trained models that aim at natural understanding or generation tasks, T5 [14] adopts an encoder-decoder framework to unify natural understanding and generation by converting the data into the text-to-text format.

#### 2.1.3 Analysis

In the development progress of pre-trained models, the critical step is the change from only attain the fixed word representations to train the whole neural architecture. The emergency of GPT introduces the paradigm of and fine-tuning making pre-trained models transferable in downstream tasks without training from scratch. The subsequent

works focus on expanding the range of text that pre-trained models can attend, improving the efficiency of models, and generalizing more downstream scenarios. For more details of models, we refer readers to [27], [28].

## 2.2 Knowledge Representation Learning

In this section, we first introduce the definition of knowledge and then the conventional methods of knowledge representation, and comprehensive knowledge representation learning based on them.

### 2.2.1 Knowledge

Knowledge is a familiarity, awareness, or understanding of someone or something, such as facts (descriptive knowledge), skills (procedural knowledge), or objects. David et al. [29] divided knowledge into four categories, namely factual knowledge, conceptual knowledge, procedural knowledge, and metacognitive knowledge. Factual knowledge refers to the knowledge of terminology and specific details and elements to describe objective things. Conceptual knowledge is the interrelationships among the fundamental elements within a larger structure that enables them to function together, such as principles, generalizations, and theories. Procedural knowledge is about the knowledge that guides action, including methods of inquiry and criteria for using skills, algorithms, techniques, and methods. Metacognitive knowledge emphasizes self-initiative and is the knowledge of cognition in general as well as awareness.

### 2.2.2 Methods of Knowledge Representation

Davis et al. [30] puts forward the definition of knowledge representation in 1993, arguing that the notion can best be understood in terms of five distinct roles. First, a knowledge representation is most fundamentally a surrogate, a substitute for the thing itself, and we can reason about the world through thinking without practice. Second, it is a set of ontological commitments about how to think about the world. Third, it is a fragmentary theory of intelligent reasoning. Fourth, it is a medium for pragmatically efficient computation, which supports recommended inferences through the effective knowledge organization. Fifth, it is the medium of human expression used to express cognition of the world.

Conventional knowledge representation methods include first-order predicate logic, frame representation [31], script representation [32], semantic network representation [33], and ontology representation. The basic grammatical elements of first-order predicate logic are symbols representing objects, relations, and functions, among which the objects refer to the individual or category of things, the relationships refer to the mapping between things, and the functions require the object in each domain to have a mapping value as a special form of a predicate. Although this method can guarantee the consistency of knowledge representation and the correctness of inference results, it is difficult to represent procedural knowledge.

A semantic network is a conceptual network represented by a directed graph where nodes represent concepts and edges represent semantic relations between concepts, which can also be transformed into triplets. It can describe knowledge in a unified and straightforward way that is beneficial for computer storage and retrieval. However, it can only represent conceptual knowledge but not dynamic knowledge such as procedure knowledge.

The framework representation organizes knowledge through a hierarchy of frames where each entity is represented by a frame containing multiple slots for storing attributes and corresponding values. It avoids duplicate definitions of the frame by inheriting one's property. Due to the diversity and complexity of the real world, many actual situations and frameworks differ greatly introducing errors or conflicts in the framework design process, which causes the lack of generality except its inability to represent procedural knowledge.

Scripted representation represents the basic behaviour of things through a series of atomic actions, which describes the occurrence of things in a definite temporal or causal order and is used for dynamic knowledge. Although it can represent procedural knowledge to a certain extent, it is not appropriate for conceptual or factual knowledge.

Originally, the term ontology comes from philosophy where it is employed to describe the existence of beings in the world. For the sake of obtaining models with reasoning capabilities, researchers adopt the term ontology to describe what can be computationally represented of the world in a program. CYC [34] is a knowledge base constructed following ontology specifications, aiming to organize human commonsense knowledge. Since ontologies can represent unanimously recognized static domain knowledge, it is also used in information retrieval and NLP. WordNet [35] is created based on word ontologies. In addition to static knowledge modelling, task-specific ontologies are also designed to add reasoning capabilities based on static knowledge.

In order to promote semantic understanding, Tim et al. [36] propose the Semantic Web concept in 2001 to build a massive distributed database that links data through semantics instead of strings. To make data understandable for computers, W3C proposes the Resource Description Framework (RDF) [37] that uses the semantic network representation to express semantics in the form of triples. This form can be easily implemented by a graph to apply graph algorithms of probability graph and graph theory to solve problems. Besides, Web Ontology Language (OWL) is designed to enable computers reasoning ability, which describes categories, attributes and instances of things complying with ontology representation.

In engineering implementation, the knowledge graph (KG) is the knowledge base represented as a network with entities as nodes and relations as edges. Specifically, the KG obtains knowledge and corresponding descriptions from the network by semantic web technology and is organized in triplets. Since the procedural knowledge is hard to manage and its certainty is weak, most of the existing KGs only contain conceptual and factual knowledge without procedural knowledge.

### 2.2.3 Knowledge Representation Learning

Knowledge representation learning (KRL) delegated by deep learning focuses on representation learning of entities and relations in the knowledge base, which effectively measures semantic correlations of entities and relations and alleviates sparsity issues. More importantly, symbolic

knowledge can be much easier to integrate with the neural network based models after knowledge representation learning.

**Translational Distance Models** With distance-based scoring functions, this type of models measure the plausibility of a fact as the distance between the two entities after a translation carried out by the relation. Inspired by linguistic regularities in [38], TransE [39] represents entities and relations in $d$-dimension vector space so that the embedded entities $h$ and $t$ can be connected by translation vector $r$, i.e., $h + r \approx t$ when $(h, r, t)$ holds. To tackle this problem of insufficiency of a single space for both entities and relations, TransH [40] and TransR [41] allows an entity to have distinct representations when involved in different relations. TransH introduces relational hyperplanes assuming that entities and relations share the same semantic space, while TransR exploits separated space for relations to consider different attributes of entities. TransD [42] argues that entities serve as different types even with the same relations and construct dynamic mapping matrices by considering the interactions between entities and relations. Owning to heterogeneity and imbalance of entities and relations, TranSparse [43] simplifies TransR by enforcing sparseness on the projection matrix.

**Semantic Matching Models** Semantic matching models measure plausibility of facts by matching latent semantics of entities and relations with similarity-based scoring functions. RESCAL [44] associates each entity and relation with a vector and matrix ,repectively. The score of a fact $(h, r, t)$ is defined by a bilinear function. To decrease the computing complexity, DistMult [45] simplifies RESCAL by restricting relation to diagonal matrices. Combining the expressive power of RESCAL with the efficiency and simplicity of DistMult, HolE [46] composes the entity representations with the circular correlation operation, and the compositional vector is then matched with the relation representation to score the triplet. Unlike models above, SME [47] conducts semantic matching between entities and relation using neural network architectures. NTN [48] combines projected entities with a relational tensor and predicts scores after a relational linear output layer.

**Graph Neural Network Models** The above models embed entities and relations by only facts stored as a collection of triplets, while graph neural network based models take account of the whole structure of the graph. Graph convolutional network (GCN) is first proposed in [49] and has been an effective tool to create node embeddings after continuous efforts [50], [51], [52], [53], which aggregates local information in the graph neighborhood for each node. As the extension of graph convolutional networks, R-GCN [54] is developed to deal with the highly multi-relational data characteristic of realistic knowledge bases. SACN [55] employs an end-to-end network learning framework where the encoder leverages graph node structure and attributes, and the decoder simplifies ConvE [56] and keeps the translational property of TransE. Following the same framework of SACN, Nathani et al. [57] propose an attention-based feature embedding that captures both entity and relation features in the encoder. Vashishth et al. [58] believe that the combination of relations and nodes should be considered comprehensively during the message transmission. Therefore they propose CompGCN that leverages various entity-relation composition operations from knowledge graph embedding techniques and scales with the number of relations to embed both nodes and relations jointly.

## 3 Overview of Knowledge Enhanced Pre-trained Models

### 3.1 The Motivation of Knowledge Enhanced Pre-trained Models

The recent progressive development of pre-trained models has attracted much attention from researchers. However, despite the great effort invested in their creation, it suffers from inability of understanding the deep semantics of text and logical reasoning. In addition to that, the knowledge learned from the model exists in parameters and is uninterpretable. Poor robustness and the lack of interpretability can be greatly alleviated by infusing entity features and factual knowledge of KGs. We name the models that integrate knowledge through retrieval or injection as KEPTMs. Most of the pre-trained models introduced in this paper focus on the leverage of linguistic knowledge and world knowledge that belongs to factual knowledge or conceptual knowledge defined in Section 2.2.1. This kind of knowledge provides rich information of entities and relations for the pre-trained model, which promotes the capability of deep understanding and reasoning of pre-trained models sharply.

### 3.2 A Taxonomy of Knowledge Enhanced Pre-trained Models

To compare and analyze existing KEPTMs, We first categorize them into three groups according to the type of injected knowledge: entity enhanced pre-trained models, triplet enhanced pre-trained models and other knowledge enhanced pre-trained models.

For entity enhanced pre-trained models, all of these models store knowledge and language information within parameters of the pre-trained model and belong to coupled-based KEPTMs. We further classify them into entity features fused and knowledge graph supervised pre-trained models according to the method of entity injection.

For triplet enhanced pre-train models, we divide them into coupled-based and decoupled-based KEPTMs by whether coupling between triplets and corpus. Since coupled-based KEPTMs entangle word embeddings and knowledge embeddings during pre-training, it fails to maintain the interpretability of symbolic knowledge. Further, we categorize coupled-based KEPTMs into three groups: embedding combined, data structure unified KEPTMs, and joint training KEPTMs according to the method of triplets infusion. As for decoupled-based KEPTMs, they preserve the embeddings of knowledge and language separately and thus introduce the interpretability of symbolic knowledge. We divide it into retrieval-based KEPTMs because it utilizes knowledge by retrieving relevant information.

Other knowledge enhanced models also can be categorized into coupled-based and decoupled-based KEPTMs. We further divide them into joint training and retrieval-based KEPTMs.

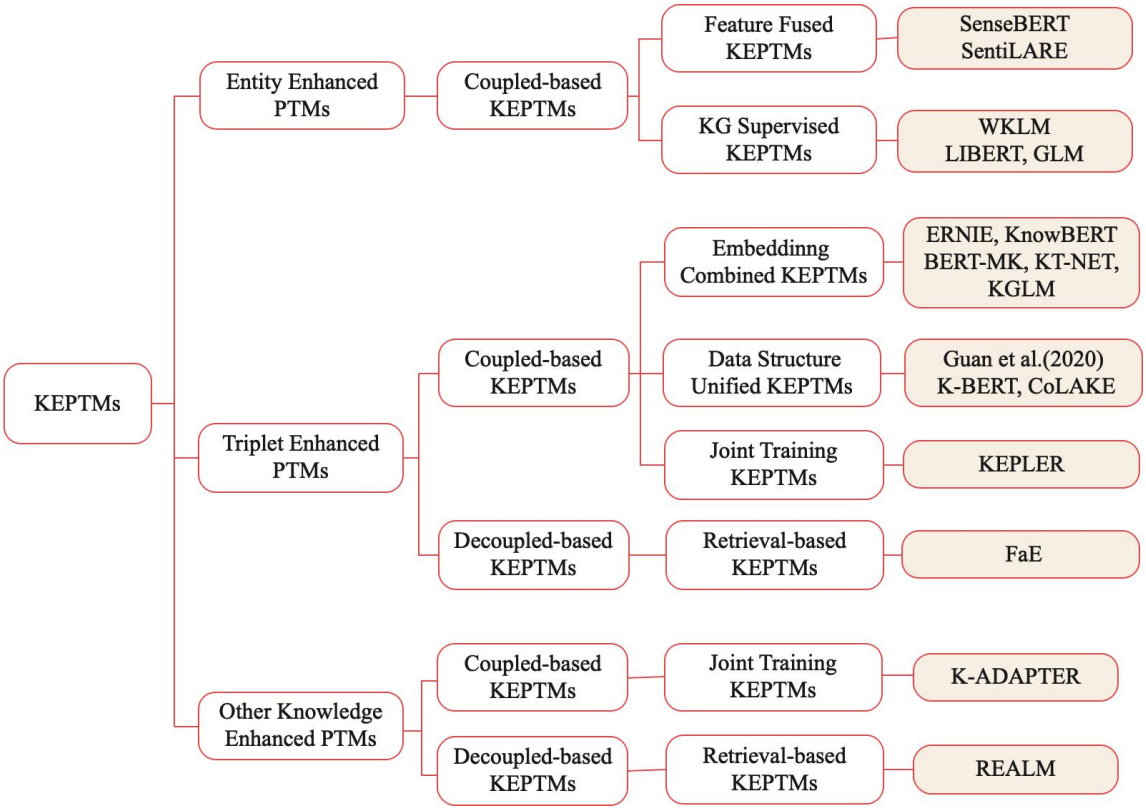



Fig. 1. Taxonomy of KEPTMs with Representative Examples.

The taxonomy and the corresponding KEPTMs introduced in the paper are shown in Fig.1.

Entity enhanced pre-trained models focus on entity-level knowledge and inject knowledge by infusing entity information and language representations. Entity features fused KEPTMs extract task-required features of entities from a KG and project them into embeddings along with sequences for pre-training, which apply for tasks accentuating semantics of entities such as sentiment analysis and word sense disambiguation. Knowledge graph supervised pre-trained models choose entities that exist a sort of relation from KGs as input data to guide models capture the implicit relations between two entities or between the chose entity and contexts, which is suitable for computing semantic similarity and knowledge completion tasks. Overall, entity enhanced pre-trained models only utilize entity information and are implemented with ease without additional network architecture and computing overhead.

Triplet enhanced pre-trained models concentrate on triplets level knowledge and inject knowledge by infusing or retrieving triplets. They benefit from using entity features and structure information of triplets, which specialize in knowledge-driven tasks such as entities and relations typing and knowledge completion. However, coupled-based KEPTMs fail to maintain interpretability of symbolic knowledge because they employ the same set of parameters to store knowledge and language information, while decoupled-based KEPTMs make the utilization of knowledge inspective and interpretable. For coupled-based KEPTMs, given the different structure of texts and triplets, different methods have been adopted to accommodate both of them and thus we introduce the following categories. Embedding combined KEPTMs infuse embeddings of sequences and knowledge after separate representation learning and alignment. Since embeddings of sequences and knowledge are learned with different algorithms, heterogeneous vector space is generated and extra work needs to be done to make it compatible. Data structure unified KEPTMs transform sequences and knowledge into a unified structure that links sequences and triplets after aligning mentions of sequences and entities in KGs. Joint training KEPTMs initialize triplets embeddings with text encoder and capture structure information by knowledge representation task. Owning to joint representation learning for sequences and knowledge, the latter two types of models project sequences and knowledge into the same vector space avoiding heterogeneous embeddings.

Rather than knowledge stored in KG, other knowledge enhanced pre-trained models leverage broader knowledge that comes from extraction from texts by mature NLP tools or external corpus. The decoupled-based model included in the first category guarantees the independence of knowledge by preserving knowledge representation separately and thus brings the interpretability for models to some extent, while the coupled-based model does not. In all, other knowledge enhanced pre-trained models achieve performance improvement by training knowledge and sequences jointly or retrieving related information from the provided corpus.

## 4 Knowledge Enhanced Pre-trained Models

### 4.1 Entity Enhanced Pre-trained Models

Instead of considering interpretability, entity enhanced pre-trained models prefer to utilize entity information and belong to coupled-based KEPTMs, which store knowledge and entity features with parameters of pre-trained models. According to different methods of entity injection, entity enhanced pre-trained models can be divided into entity features fused and knowledge graph supervised KEPTMs. The former, like SenseBERT [59] and SentiLARE [60], infuses representations of entity features and word embedding by text encoder with the guidance of pre-training tasks, and the latter, like WKLM [61], LIBERT [62] and GLM [63], exploits the power of encoder to capture implicit knowledge of texts under the supervision of KGs.

#### *4.1.1 Entity Feature Fused KEPTMs*

SenseBERT [59] infuses word-sense information into BERT's pre-training signal, which enhances the ability of lexical understanding and thus solves the problem that BERT cannot well learn representations of rare words affected by heavy-tailed distribution. Following BERT architecture, jointly with the standard MLM, SenseBERT trains a semantic-level language model (SLM) that predicts the missing words meaning. We illustrate the framework of SenseBERT in Fig. 2.

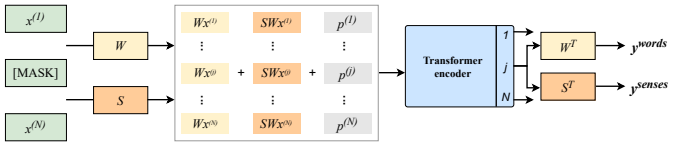



Fig. 2. The Framework of SenseBERT [59].

SenseBERT takes a sequence with masked words as input and feeds it into Transformer block after projecting word information and its supersenses to the embeddings. After that, the model are pre-trained in word-form and word sense tasks with loss functions formulated as:

$$\mathcal{L}_{MLM} = -\log P(w|context), \tag{2}$$

$$\mathcal{L}_{SLM} = -\log \sum_{s \in A(w)} P(s|context), \tag{3}$$

repestively. Where $w$ is the masked word and $s$ denotes the possible supersense.

Without compromising performance in General Language Understanding Evaluation (GLUE) [64], SenseBERT boosted word-level semantic awareness considerably outperforms a vanilla BERT on a Supersense Disambiguation task and achieves state of the art results on the Word in Context task [65].

Although BERT has been proved successful in simple sentiment classification, directly applying it to fined-grained sentiment analysis shows less significant improvement [66]. Therefore, to better solve the above issue, SemtiLARE [60] is proposed to inject sentiment polarity and its part-of-speech for BERT by label-aware MLM objective.

First, SemtiLARE acquires the part-of-speech tag $pos_i$ of each word $x_i$ via Stanford Log-Linear Part-of-Speech Tagger[1] and computes words sentiment polarity $polar_i$ from SentiWordNet (SWN) by context-aware attention mechanism. Given the knowledge enhanced text sequence $X_k = \{(x_i, pos_i, polar_i)\}_i^n$, SemtiLARE takes RoBERTa as the backbone model and feeds the summed embeddings $\hat{X}_k$ of $X_k$ into the encoder. During the pre-training phase, early infusion (EF) and late supervision (LS) are utilized to capture the relationship between sentence-level language representation and word-level linguistic knowledge. EF aims to recover the masked sequence conditioned on the sentence-level label while LS simultaneously predicts the sentence-level sentiment label, words, and linguistic knowledge of words. The loss of two subtasks is computed with the following functions:

$$\mathcal{L}_{EF} = -\sum_{t=1}^{n} m_t \cdot \Big[\log P(x_t|\hat{X}_k, l) + \log P(pos_t|\hat{X}_k, l) + \log P(polar_t|\hat{X}_k, l)\Big], \tag{4}$$

$$\mathcal{L}_{LS} = -\log P(l|\hat{X}_k) - \sum_{t=1}^{n} m_t \cdot [\log P(x_t|\hat{X}_k) + \log P(pos_t|\hat{X}_k) + \log P(polar_t|\hat{X}_k)], \tag{5}$$

where $l$ is additional sentence sentiment annotation.

Comparing to baselines [11], [12], [13], [67], [68], [69], SentiLARE refreshes the state of the art performance of language representation models on both sentence- and aspect-level sentiment analysis tasks on dataset [70], [71], [72], [73], [74], [75], and thus facilitates sentiment understanding.

#### *4.1.2 Knowledge Graph Supervised Pre-trained Models*

In addition to eliciting entity features from the KGs, researchers also choose the entities as training data under the supervision of KGs.

To directly derive real-world knowledge from unstructured text, WKLM [61] designs the weakly supervised entity replacement detection (ERD) training objective to explicitly force the model to incorporate knowledge about real world entities.

Specifically, it first recognizes the entity mentions and links them to Wikipedia entities and then replaces mentions with other entities of the same type of linked entities after consulting KGs. Following BERT architecture, the model employs entity replacement and MLM objective in a multi-task training strategy. Given a certain entity $e$ mentioned in a context $C$, the binary prediction loss function $\mathcal{L}$ for entity replacement objective is formulated as:

$$\mathcal{L} = \mathbb{I}_{e \in \varepsilon^+} \log P(e|C) + (1 - \mathbb{I}_{e \in \varepsilon^+}) \log(1 - P(e|C)). \tag{6}$$

Compared to the MLM objective, the entity replacement task introduces stronger entity level negative signals and preserves the linguistic correctness of the original sentence.

WKLM is evaluated in downstream tasks and consistently outperforms BERT with an average 2.7% $F_1$ improvement on entity-related question answering datasets [76], [77], [78], [79] and 5.7% accuracy improvement on fine-grained entity typing dataset [80] .

Instead of using a single entity, LIEBRT [62] takes entity pairs meeting semantic similarity constraints as additional

1. http://nlp.stanford.edu/software/tagger.html

training instances to enable BERT to understand the lexical-semantic relations. The framework of the model is shown in Fig. 3.

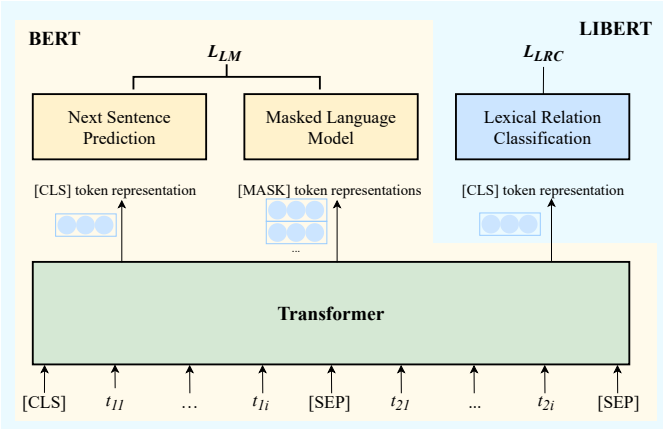



Fig. 3. The Framework of LIBERT [62].

Concretely, synonyms pairs and direct hyponym-hypernym pairs are selected from WordNet [35] and used as positive training examples, while negative examples are created by replacing one entity of pairs. After transforming each instance into a sequence of tokens, LIBERT is trained in multitask learning setting where it couples BERT's MLM and NSP objectives with an auxiliary task of lexical relation classification (LRC). The gains of lexical knowledge injection are observed for 9 out of 10 language understanding tasks from the GLUE benchmark, and for 3 lexical simplification benchmarks.

Different from leveraging lexical-similar entity pairs, GLM [63] drives pre-trained models to capture implicit relations underlying raw text between related entities through the guidance of a KG.

During data preparation, GLM selects informative entities as with a linked KG and then masks their corresponding mentions after aligning mentions and entities. The selected probability $P(e_j)$ of an entity $e_j$ is determined by its frequency in a document and the length of the shortest undirected path between the two entities over the linked KG, which avoids masking trivia and undeducible entities. $P(e_j)$ is given by:

$$P(e_j) \propto \mathbb{I}_{\{DF(e_j)<R_{thresh}\}} \times [|NB(e_j)|]_{R_{min}}^{R_{max}}, \tag{7}$$

where $NB(e) \triangleq \{e'|PLen(e' \leftrightarrow e) < R_{hop} \wedge e' \in \varepsilon\}$, $PLen(e' \leftrightarrow e)$ denotes the length of the shortest undirected path between two entity, $\varepsilon$ is the set of candidate entities, the term $DF(\cdot)$ denotes document frequency and $[x]_a^b \triangleq \max(a, \min(x, b))$.

With qualified masked entities as positive samples, negative entity samples are derived from the knowledge graph and used as distractors for the masked entities to make the learning more effective. Then the entity level MLM task and distractor-suppressed ranking (DSR) task are harnessed to train the model, which is used for predicting the mask entity and distinguish the positive sample from the negative one of entity pairs, respectively.

The total loss function is determined by summing loss of two pre-training tasks with the latter weighted by a hyperparameter $\gamma$:

$$\mathcal{L} = \mathcal{L}_{MLM} + \gamma \mathcal{L}_{DSR}. \tag{8}$$

After fine-tuning, GLM achieves improved performance on two benchmark datasets [81], [82] for question answering and three datasets [83], [84] for knowledge base completion tasks.

## 4.2 Triplet Enhanced Pre-trained Models

Triplet enhanced pre-trained models include coupled-based and decoupled-based KEPTMs. The former preserves triplet and text information in parameters of pre-trained models and thus loses interpretability while the latter preserves them separately and retains interpretability of symbolic knowledge. In this section, we first introduce coupled-based KEPTMs according to different methods of knowledge infusion and then a decoupled-based model.

### *4.2.1 Coupled-based KEPTMs*

**Embedding Combined KEPTMs** For accommodating different structure of text and triplets, embedding combined KEPTMs learn both representations separately and infuse their embeddings by attention mechanism.

To make full use of lexical, syntactic, and knowledge information simultaneously, ERNIE [85] integrates entity embeddings pre-trained on triplets with corresponding entity mentions in the text. The framework of ERNIE is shown in Fig. 4.

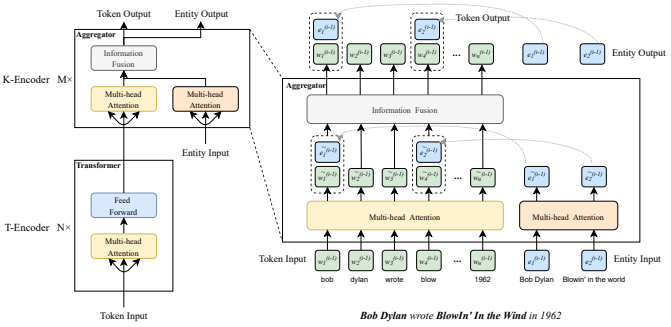



Fig. 4. The Framework of ERNIE [85].

ERNIE encoder consists of a text encoder (T-encoder) for early representation learning for text and a knowledge encoder (K-encoder) to integrate texts and entities. Before pre-training, entity mentions in the text are firstly recognized and then aligned to corresponding entities in KGs. ERNIE encodes the entities and relations with knowledge representation learning algorithm (e.g, TransE) and integrates entity representations and token embeddings in K-encoder based on the alignments. In addition to MLM and NSP tasks of BERT, a denoising entity auto-encoder (DEA) objective is designed to predict entities with integrated embeddings. Given the token sequence $\{w_1, \cdots, w_n\}$ and corresponding entity sequence $\{e_1, \cdots, e_m\}$, ERNIE defines the aligned entity distribution for the token $w_i$ as follows,

$$p(e_j|w_i) = \frac{\exp(linear(w_i^o) \cdot e_j)}{\sum_{k=1}^{m} \exp(linear(w_i^o) \cdot e_k)}, \tag{9}$$

where $linear(\cdot)$ denotes a linear layer.

The better performance of ERNIE in entity and relation classification task demonstrates the effectiveness of triplets infusion. The comparable results with BERT on GLUE illustrate ERNIE does not lose the textual information.

Similarly, KnowBERT [86] trains BERT jointly with an entity linking model to incorporate entity representation in an end-to-end fashion, which complements factual knowledge for BERT. The process of entity linking and integration is shown in Fig. 5.

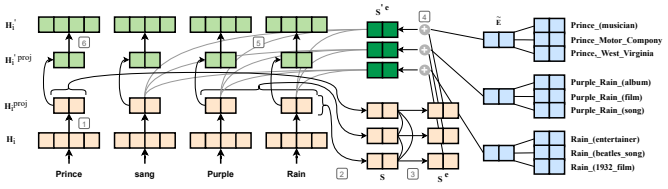



Fig. 5. The Process of Entity Linking and Integration [86].

Given input sequences, the model recognizes entity mentions and projects it to $H_i^{\prime proj}$ in stage (1), and then mention-span representations are computed in stage (2). To achieve entity disambiguation, the model performs a self-attention for mention-span representations in stage (3). An integrated entity linker retrieves relevant entity embeddings pre-trained by algorithm TuckER [87] and computes weighted average entity embeddings $\widetilde{E}$. After that, the model infuses mention representations and entities via a form of word-to-entity attention in stage (4). Then the integrated representations are recontextualized with word-to-entity-span attention to deliver knowledge to the whole sequence in stage (5), and are projected back to the BERT dimension $H_i'$ in the last stage.

Notably, entity linking (EL) is added as an auxiliary pre-training task to link mentions in the sequences and entities of KG. Comparing to ERNIE, although KnowBERT needs to train entity linking model with additional annotated labels, it considers more related entities and thus injects richer knowledge. After integrating WordNet and a subset of Wikipedia into BERT, KnowBERT demonstrates improved ability to recall facts as measured in a probing task and knowledge-driven tasks like relationship extraction, entity typing, and word sense disambiguation.

So far, all models introduced treats triplets as a training unit during KRL procedure, ignoring the connection of triplets. BRET-MK [88] captures richer semantic of triplets from KG by utilizing contextualized information of the nodes. Following ERNIE, the knowledge embeddings are pre-trained with TransE. After alignment, the subgraphs of entities are extracted from the KG and transformed into a sequence, which is shown in Fig. 6. Considering mutual influence of entities and relations, the relations are regarded as graph nodes as well. Then the sequence of nodes is fed into Transformer to further encode contextual information of entity by knowledge embedding objective defined in (10).

$$\mathcal{L} = \sum_{t \in T} \max\{d(t) - d(f(t)) + \gamma, 0\}, \tag{10}$$

where $t = (t_s, t_r, t_o)$, $d(t) = |t_s + t_r - t_o|$, $f(t)$ denotes random replacement of head or tail entity in a valid triplet

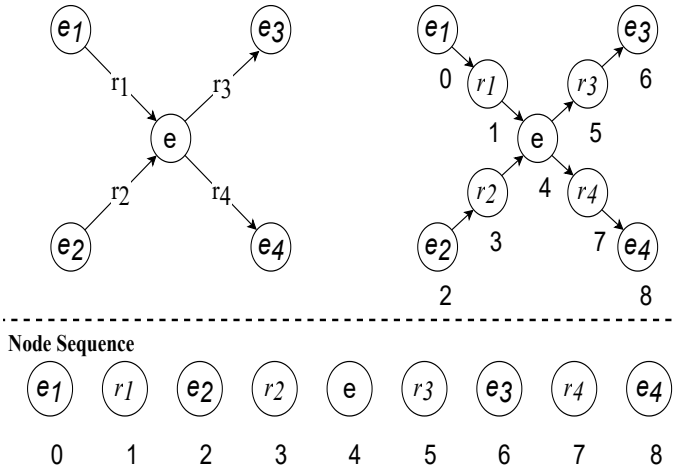



Fig. 6. The Conversion of the Subgraph [88].

and $\gamma > 0$ is a margin hyperparameter. After that, the same knowledge integration framework with ERNIE is utilized.

Pre-trained with medical corpus and KG, BERT-MK outperforms BERT-Base, BioBERT [89], and SCIBERT [90] by 0.71%, 0.24%, and 0.02% on the average accuracy in NER, respectively. In the relations classification task, BERT-MK improves the average $F_1$ score of all dataset [91], [92] at least by 2.27% comparing to baselines [93], [94].

Following BERT architecture, KT-NET [95] employs an attention mechanism to select desired knowledge from KGs adaptively and then fuses the selected knowledge with BERT to enable knowledge- and context-aware predictions for machine reading comprehension. Like the above models, triplets in KG are first encoded by KRL, namely BILINEAR [45] in KT-NET. Then, given a question and a passage, KT-NET employs a BERT encoding layer to compute context-aware embeddings for the reading text and integrates entity embeddings with context representations after retrieving potentially relevant entity from WordNet and NELL [96]. As concepts in KG are not necessarily relevant to the token, a knowledge sentinel is introduced to avoid attending trivia entities. Then the self-match layer, the core component, fuses both representations to enable rich interactions among them.

With output integrated token embedding $o_i$, the probability of each token to be the start or end position of the answer span and the answer boundary prediction (ABP) loss function $\mathcal{L}$ are formulated as follow:

$$P_i^1 = \frac{\exp(w_1^\top o_i)}{\sum_j \exp(w_1^\top o_j)}, P_i^2 = \frac{\exp(w_2^\top o_i)}{\sum_j \exp(w_2^\top o_j)}, \tag{11}$$

$$\mathcal{L} = -\frac{1}{N} \sum_{j=1}^{N} (\log p_{y_j^1}^1 + \log p_{y_j^2}^2), \tag{12}$$

respectively. Where $w_1, w_2$ are trainable parameters, $N$ is the number of examples in the dataset and $y_j^1, y_j^2$ are the true start and end positions of the j-th example, respectively.

The results in question answering indicate that KT-NET offers significant and consistent improvements over BERT and outperforms other competitive baselines [97], [98], [99], [100] on ReCoRD [97] and SQuAD1.1 [101] benchmarks.

To supply factual knowledge for the generation task, KGLM [102] is constructed to render information from a local KG that is dynamically built by selecting and copying facts based on context from an external KG.

Inheriting LSTM [22] network architecture, KGLM encodes input sequences with a recurrent structure and splits the hidden state $h_t$ into three components: $h_t = [h_{t,x}; h_{t,h}; h_{t,r}]$ where $h_{t,x}$ decides the source of the next word among the existing local KG, the vocabulary and external KG. $h_{t,h}$ and $h_{t,r}$ are used to further choose the relevant entity as the generative word from the external KG. The chosen probability $P(e_t)$ of the entity $e_t$ is determined by: $P(e_t) = \text{softmax}(v_e \cdot (h_{t,h} + h_{t,r})$ where $v_e$ denotes entity embeddings pre-trained by TransE. Once the entity is chosen, the corresponding triplets will be added to the local knowledge graph to broaden the scope of candidate references. However, if the source of the next word is the local KG, the generative word needs to be selected from the local KG. After pre-trained with language modeling (LM) objective, KGLM is evaluated using perplexity of held-out corpus and accuracy on fact completion. The results show that KGLM attains substantially lower perplexity than AWD-LSTM [103] and ENTITYNLM [104], and achieves the highest accuracy on fact completion comparing to GPT-2 and AMD-LSTM.

**Data Structure Unified KEPTMs** Due to separate representation learning for text and knowledge, embedding fused pre-trained KEPTMs suffer from the infusion of heterogeneous vector spaces. To avoid it, data structure unified KEPTMs link triplets with text and form a unified format before training.

To generate reasonable stories with commonsense knowledge, Guan et al. [105] transform triplets into the format of texts and train the model based on GPT by multitask learning. Concretely, the model transforms the commonsense triples in ConceptNet and ATOMIC into readable natural language sentences using a template-based method [106] and carries out post-training with these sentences by LM objective.

In addition, the classification task (CLS) is adopted to distinguish true stories from auto-constructed fake stories during fine-tuning, which makes the model implicitly capture the causal, temporal dependencies between sentences and inter-sentence coherence.

The automatic and manual evaluation show that the model can generate more reasonable stories than state of the art baselines [107], [108], [109], [110], [111], [112], particularly in logic and global coherence.

The above model treats text and triplets as independent training data and injects knowledge during pre-training, while K-BERT [113] connects sequences with relevant triples by constructing the knowledge-rich sentence tree to achieve knowledge injection during fine-tuning.

Specifically, all the entity mentions involved in the sentence are selected out to query corresponding triples in KGs, and then K-BERT stitches the triples to corresponding positions to generate a sentence tree shown in Fig. 7. To avoid interfering in text encoding, the model adopts soft-position and visible matrix to control the scope of knowledge. Taking BERT as the backbone model, K-BERT exploits suitable KG for different scenarios and fine-tuned parameters on open

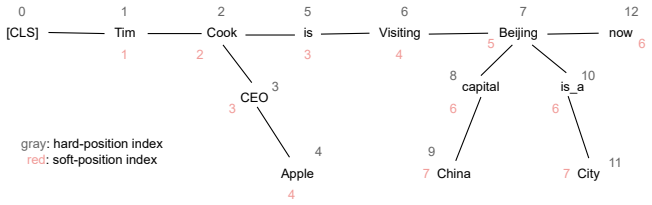



Fig. 7. The Structure of the Sentence Tree [113].

and specific domain downstream tasks with pre-training tasks of BERT.

K-BERT outperforms BERT in all open domain tasks except sentiment analysis and achieves 1-2% $F1$ gains in specific domain tasks. It is worth mentioning that K-BERT fine-tuned with CN-DBpedia [114] performs better than that with HowNet [115] in question answering and named entity recognition (NER) while the latter gains further improvement in semantic similarity tasks, which demonstrates the importance of an appropriate KG for different scenarios.

Although K-BERT infuses triplets and sequences by unifying data structure, it is unsure whether the model captures the explicit relations between entities.

Different from the model proposed by Guan et al. and K-BERT, CoLAKE [116] integrates contextual triplets instead of independent triplets by constructing a word-knowledge graph. Concretely, CoLAKE transforms tokenized sequence into a fully connected graph and elicits a subgraph for each entity in the sequence that contains the triplets about the entity. After that, the word-knowledge graph is built by replacing mentions in fully connected graph with the aligned entity. The process is shown in Fig. 8. Following K-BERT, the visible matrix and soft-position are employed to control the scope of encoding.

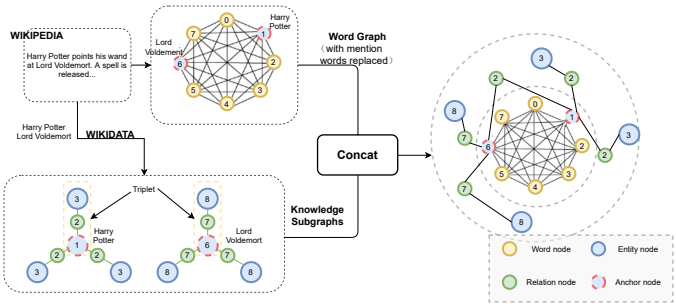



Fig. 8. The Construction of Word-knowledge Graph [116].

Pre-trained with RoBERTa network architecture and MLM objective, CoLAKE is evaluated in knowledge-driven and language understanding tasks. It achieves the highest $F_1$ score and outperforms 0.8% and 0.3% than ERNIE and KnowBERT in the entity classification task, respectively. In the relation extraction task, it achieves at least 2% gains in terms of recall, accuracy and $F_1$ than all baselines. The results in GLUE are comparable with RoBERTa, which demonstrates that CoLAKE maintains the ability of language understanding.

**Joint Training KEPTMs** Distinguish from the idea of unified structure, KEPLER [117] jointly optimizes parameters with the KE and MLM objectives to blend factual

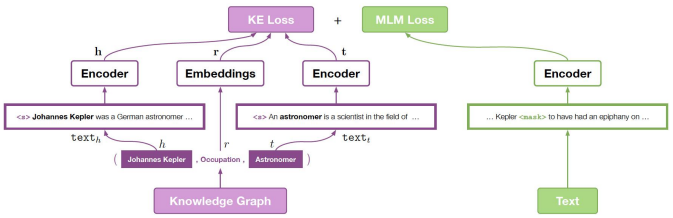



Fig. 9. The Framework of KEPLER [117].

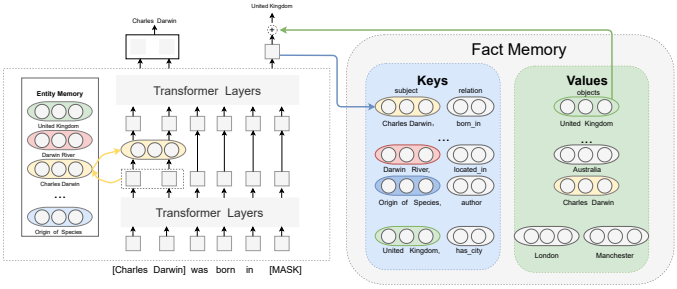



Fig. 10. The Framework of FaE [118].

knowledge with language representations. The infusion process is shown in Fig. 9.

Instead of using pre-trained knowledge embeddings by KRL, KEPLER chooses RoBERTa as the base model and initializes knowledge embeddings with textual descriptions, which avoids the appearance of heterogeneous vector space. Given the textual descriptions elicited from Wikipedia, three kinds of textual data are employed to generate initial embeddings of entities and relations for KE objective: entity descriptions as embeddings, entity and relation descriptions as embeddings, and entity embeddings conditioned on relations. Taking the last textual data as example, KEPLER encodes embeddings of head entities with the concatenation of the descriptions for head entities and relations, embeddings of tail entities only with its description, and initializes relation embeddings randomly. After that, the KE objective is designed to incorporate structure information of factual knowledge into language representations. In addition to KE, MLM is also adopted to update parameters in a multitask setting. The total loss of the model is determined as:

$$\mathcal{L} = \mathcal{L}_{KE} + \mathcal{L}_{MLM}. \tag{13}$$

The experimental results in relation classification show that KEPLER-Wiki, a KEPLER variant equipped with Wikidata, outperforms 1.1% $F1$ than KnowBERT and 1.3% $F1$ than RoBERTa and ERNIE. For entity typing, KEPLER-Wiki outscores 0.6% than KnowBERT and 1.3% than ERNIE in terms of $F1$. Compare to models trained only with KRL, KEPLER achieves the highest accuracy in linking prediction under inductive setting with language representations.

#### 4.2.2 Deoupled-based KEPTMs

Coupled-based KEPTMs focus on integrating knowledge into pre-trained models, while decoupled-based KEPTMs leverage knowledge by retrieving what it needs and thus introduce interpretability of knowledge.

**Retrieval-based KEPTMs** FaE [118] designs an explicit interface based on a neural language model to connect symbolically interpretable factual information and language representations, which achieves inspection and interpretation of knowledge. Owning to the decoupling of knowledge representations and language representations, FaE can change the output of the language model by modifying only the non-parametric triplets without any additional training. The framework of FaE is showed in Fig. 10. Specifically, FaE includes an additional memory called a fact memory, which encodes triples from a symbolic KG and transforms it into the form of key-value to retrieve tail entities effectively. Given a paragraph and its mentions, the model constructs input as a cloze-type question answering task by masking a mention. Then input sequences are encoded as embeddings following the BERT network architecture and knowledge embeddings are initialized randomly. On the one hand, FaE learns to jointly link entities from unmasked mentions using entity-aware contextual embeddings, which is implemented by entity linking (EL) and entity prediction (EP) pre-training tasks. On the other hand, the model predicts the masked entity using knowledge enhanced embeddings, which is implemented by distant supervision (SR) and entity-level MLM tasks. The total loss of the model is defined as:

$$\mathcal{L} = \mathcal{L}_{EL} + \mathcal{L}_{EP} + \mathcal{L}_{SR} + \mathcal{L}_{MLM}. \tag{14}$$

Pre-trained on Wikipedia and Wikidata, FaE is evaluated in FreebaseQA [119] and WebQuestionSP [120], further split into full and Wikidata answerable settings. The results show that FaE outperforms by nearly 10 points than other baselines [119], [121], [122] in FreebaseQA. Despite the model obtains relatively lower performance in the full dataset setting due to incompleteness of referential KG, it achieves the highest accuracy in the Wikidata answerable setting of WebQuestionSP.

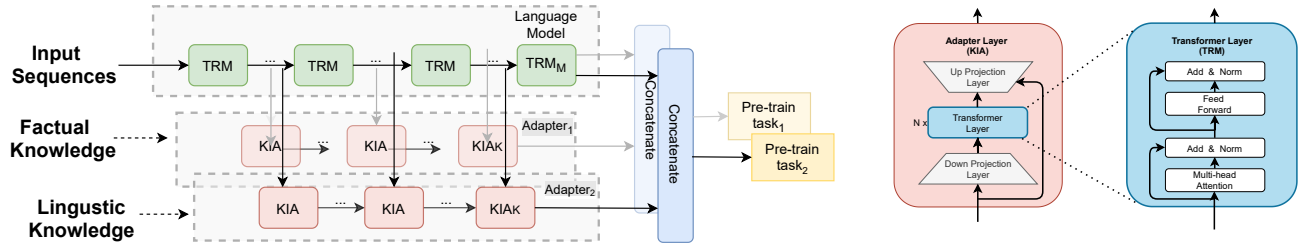
### 4.3 Other Knowledge Enhanced Pre-trained Models

Instead of elements in KGs, other knowledge enhanced pre-trained models explore more coarse-grained knowledge from the external corpus or extracted by existing NLP tools.

#### 4.3.1 Coupled-based KEPTMs

**Joint Training KEPTMs** With the same knowledge injection method of KEPLER, K-ADAPTER [123] also updates parameters by jointly learning knowledge and language information. The difference is that based on RoBERTa, K-ADAPTER designs an adapter for storing each kind of infused knowledge to keep original parameters of the pre-trained model fixed and isolate the interaction of different knowledge, which addresses the issue of catastrophic forgetting. The framework of K-ADAPTER is shown in Fig. 11.

Plugged among different transformer layers of the pre-trained model, the adapter concatenates the output of the transformer layer and the previous adapter as input. To obtain factual and linguistic knowledge, K-ADAPTER aligns Wikipedia text to Wikidata triplets and extracts dependency relations by applying off-the-shell dependency parser to web texts. Then the predication classification (PC) task and dependency relation prediction (DRP) task are utilized in multitask setting to infuse knowledge.




(a) The Process of Knowledge Infusion

(b) The Structure of the KIA

Fig. 11. The Framework of K-ADAPTER [123].

K-ADAPTER is evaluated on entity typing, question answering and relation classification. For entity typing, it achieves an improvement of 1.38% $F_1$ over RoBERTa that performs best among baseline models in Open Entity, and improves the macro $F_1$ by 2.88%, micro $F_1$ by 2.54%, and accuracy by 1.60% than WKLM [61] in FIGER. For question answering, it significantly achieves 11.89% improvement of accuracy than BERT-FT [124], 1.24% than RoBERTa in CosmoQA [124], and 3.1% than WKLM in Quasar-T [78]. As for relation classification, it significantly outperforms all baselines and achieves 0.79%, 0.54%, and 0.34% improvement than RoBERTa, KnowBERT, and KEPLER, respectively.

#### 4.3.2 Decoupled-based KEPTMs

**Retrieval-based KEPTMs** To capture knowledge in a more modular and interpretable way, REALM [125] augments the language model by pre-training with a latent knowledge retriever, which allows the model to retrieve and attend over documents from a large corpus. Specifically, it consists of two key components: the neural knowledge retriever implemented with BERT framework, which encodes input data and retrieves possibly helpful documents, and the knowledge-augmented encoder implemented with a Transformer, which is used to infuse entities in documents and predicts words for question answering.

During pre-training, given a sentence $x$ with some masked words, the model first retrieves relevant documents $z$ from a knowledge corpus $\mathcal{Z}$ by learning the distribution $P(x|z)$ and then predicts those missing tokens $y$ by learning the distribution $P(y|x,z)$ with MLM task. For fine-tuning, the task is an open domain question answering (Open-QA): $x$ is a question, and $y$ is the answer. To avoid retrieving unrelated documents in stage of retrieval, the model employs inverse cloze task (ICT) [126] to warm-start the embeddings of document and query.

Evaluated on three different datasets [125], [127], [128], REALM outperforms all previous approaches [10], [14], [76], [129], [130], [131] by a significant margin and achieves a 3.3% accuracy improvement than T5 (11B) [14] that gets the best result among all baseline models while being 30 times smaller.

We illustrate all introduced KEPTMs more details in Table 1.

## 5 Conclusion and Future Directions

We analyze and compare the existing KEPTMs from three perspectives: the type of knowledge, the coupling between knowledge and text, and the method of knowledge injection in a hierarchical manner. Entity enhanced pre-trained models utilize entity information without introducing additional network and computational overhead, which are simple to implement and suitable for tasks requiring fine-grained entity features. Despite of more efforts, triplet enhanced pre-trained models can inject both entity and relation features and generalize knowledge-driven tasks like entity classification, relationship extraction and knowledge completion. Other knowledge enhanced pre-trained models provide the novel ideas of leveraging other forms of knowledge and makes existing NLP tools and corpus applicable for more scenarios.

Coupled-based KEPTMs focus on training the model transferable to language understanding and knowledge-driven NLP tasks, while sacrifice interpretability. Among them, joint training KEPTMs achieve knowledge infusion with minimum work, which is implemented by designing an appropriate pre-training task. Decoupled-based KEPTMs help us understand how pre-trained models harness knowledge for downstream tasks and provide a guide for better usage and further improvement.

Although KEPTMs have proven their power for various NLP tasks, challenges still exist due to the complexity of knowledge and language. We suggest following four future directions for KEPTMs.

(1) Based on semantic network representation, triplets have become the most popular form to organize knowledge. However, as we have analyzed, more works have to be done for heterogeneous infusion between triplets and sequences. Besides semantic network representation, there are a multitude of knowledge representation methods presenting properties of knowledge in different forms. Therefore, searching an appropriate knowledge representation for different knowledge is promising.

(2) Entity enhanced KEPTMs and triplet enhanced KEPTMs utilize entity and relation information in KG, ignoring the topological structure of connected entities. The topological structure information can be modelled by graph representation learning. Therefore, infusing graph structure information into language representations will be a valid direction.

Besides exploring richer information of KG, other types of knowledge is also worthy of being considered. So far, the KEPTMs we introduce focus on injecting factual or conceptual knowledge. However, procedural and metacognitive knowledge also play an significant role in reasoning

TABLE 1
List of Representative KEPTMs

| KEPTMs | Pre-trained Model | Pre-training Task | KRL | Corpus | Type of Knowledge | Method of Injection | Explain? |
|---|---|---|---|---|---|---|---|
| SenseBERT | BERT | MLM<br>SLM | \ | SemCor<br>WordNet | Entity | Feature Combined | No |
| SentiLARE | RoBERTa | EF<br>LS | \ | Yelp*<br>SWN | Entity | Feature Combined | No |
| WKLM | BERT | MLM<br>ERD | \ | WikiEn<br>Wikidata | Entity | Supervision of KG | No |
| LIBERT | BERT | MLM<br>NSP<br>LRC | \ | WikiEn<br>WordNet | Entity | Supervision of KG | No |
| GLM | BERT or RoBERTa | MLM<br>DSR | \ | WikiEn<br>ConceptNet | Entity | Supervision of KG | No |
| ERNIE | BERT | MLM<br>NSP<br>DEA | TransE | WikiEn<br>Wikidata | Triplets | Feature Combined | No |
| KnowBERT | BERT | MLM<br>NSP<br>EL | TuckER | WikiEn<br>Wikidata<br>WordNet | Triplets | Feature Combined | No |
| BERT-MK | BERT | MLM<br>NSP<br>DEA<br>KE | TransE | PubMed<br>UMLS | Triplets | Feature Combined | No |
| KT-NET | BERT | ABP | BILINEAR | ReCoRD<br>SQuAD1.1<br>NELL<br>WordNet | Triplets | Feature Combined | No |
| KGLM | LSTM | LM | TransE | Linked-WikiText-2<br>Wikidata | Triplets | Feature Combined | No |
| Guan et al. | GPT-2 | LM<br>CLS | \ | ROCStories<br>ConceptNet<br>ATOMIC | Triplets | Unified Data Structure | No |
| K-BERT | BERT | MLM<br>NSP | \ | WikiZh<br>WebtextZh<br>CN-DBpedia<br>HowNet<br>MedicalKG | Triplets | Unified Data Structure | No |
| CoLAKE | RoBERTa | MLM | \ | WikiEn<br>Wikidata-5M | Triplets | Unified Data Structure | No |
| KEPLER | RoBERTa | MLM<br>KE | \ | WikiEn<br>Wikidata<br>WordNet | Triplets | Joint Training | No |
| FaE | BERT | MLM<br>EL<br>EP<br>DS | \ | WikiEn<br>Wikidata | Triplets | Triplet Retrieval | Yes |
| K-ADAPTER | RoBERTa | DRP<br>PC | \ | WikiEn<br>Web texts | Linguistic and factual knowledge | Joint Training | Yes |
| REALM | BERT | MLM<br>ICT | \ | WikiEn<br>CC-news | Document | Document Retrieval | Yes |

* Yelp refers to Yelp Dataset Challenge 2019.

and judgment for the open world. Thus, a more attractive direction is to explore the infusion between two types of knowledge above and corpus.

(3) Although KEPTMs reach progressive performance, the deep non-linear architecture of the pre-trained model makes the procedure of decision-making uninterpretable. Symbolic knowledge can bring interpretability with explicit semantic. Retrieval-based KEPTMs set valuable examples of introducing interpretability while injecting knowledge into pre-trained models. Designing a knowledge injection method without destructing inspection of symbolic knowledge will significantly improve interpretability.

(4) Text- and image-based multi-modal models capture rich semantics in the image and associated text by learning image-text representations and have been applied in captioning, visual question answering and visual reasoning tasks. However, the image features learned cannot capture detailed semantics depicted in an image. Moreover, the pre-training of multi-modal models usually rely on a assumption that there is a strong correlation between text data and image data.The utilization of well-organized knowledge for multi-modal models need to be explored to break the limitation of this assumption and supply rich semantics of images.

## REFERENCES


[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[2] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1, no. 2.

[3] Y. Bengio, *Learning deep architectures for AI*. Now Publishers Inc, 2009.

[4] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[5] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in Neural Information Processing Systems*, vol. 26, pp. 3111–3119, 2013.

[6] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[7] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[8] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018, pp. 2227–2237.

[9] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training."

[10] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186.

[12] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," *Advances in Neural Information Processing Systems*, vol. 32, pp. 5753–5763, 2019.

[13] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[14] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, pp. 1–67, 2020.

[15] L. Dong, N. Yang, W. Wang, F. Wei, X. Liu, Y. Wang, J. Gao, M. Zhou, and H.-W. Hon, "Unified language model pre-training for natural language understanding and generation," *arXiv preprint arXiv:1905.03197*, 2019.

[16] H. Bao, L. Dong, F. Wei, W. Wang, N. Yang, X. Liu, Y. Wang, J. Gao, S. Piao, M. Zhou *et al.*, "Unilmv2: Pseudo-masked language models for unified language model pre-training," in *International Conference on Machine Learning*. PMLR, 2020, pp. 642–652.

[17] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," in *International Conference on Learning Representations*, 2020.

[18] K. Song, X. Tan, T. Qin, J. Lu, and T.-Y. Liu, "Mass: Masked sequence to sequence pre-training for language generation," in *International Conference on Machine Learning*. PMLR, 2019, pp. 5926–5936.

[19] S. Ji, S. Pan, E. Cambria, P. Marttinen, and S. Y. Philip, "A survey on knowledge graphs: Representation, acquisition, and applications," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[20] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[21] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A neural probabilistic language model," *The journal of machine learning research*, vol. 3, pp. 1137–1155, 2003.

[22] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[23] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 328–339.

[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *NIPS*, 2017.

[25] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners."

[26] R. Kiros, Y. Zhu, R. Salakhutdinov, R. S. Zemel, A. Torralba, R. Urtasun, and S. Fidler, "Skip-thought vectors," in *Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 2*, 2015, pp. 3294–3302.

[27] Q. Liu, M. J. Kusner, and P. Blunsom, "A survey on contextual embeddings," *arXiv preprint arXiv:2003.07278*, 2020.

[28] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang, "Pre-trained models for natural language processing: A survey," *Science China Technological Sciences*, pp. 1–26, 2020.

[29] D. R. Krathwohl, "A revision of bloom's taxonomy: An overview," *Theory into practice*, vol. 41, no. 4, pp. 212–218, 2002.

[30] R. Davis, H. Shrobe, and P. Szolovits, "What is a knowledge representation?" *AI magazine*, vol. 14, no. 1, pp. 17–17, 1993.

[31] M. Minsky, *A framework for representing knowledge*. de Gruyter, 2019.

[32] S. S. Tomkins, "Script theory: Differential magnification of affects." in *Nebraska symposium on motivation*. University of Nebraska Press, 1978.

[33] M. R. Quillan, "Semantic memory," BOLT BERANEK AND NEWMAN INC CAMBRIDGE MA, Tech. Rep., 1966.

[34] D. Lenat and R. Guha, "Building large knowledge-based systems: Representation and inference in the cyc project," *Artificial Intelligence*, vol. 61, no. 1, p. 4152, 1993.

[35] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[36] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific american*, vol. 284, no. 5, pp. 34–43, 2001.

[37] E. Miller, "An introduction to the resource description framework," *Bulletin of the American Society for Information Science and Technology*, vol. 25, no. 1, pp. 15–19, 1998.

[38] T. Mikolov, W.-t. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations," in *Proceedings of the 2013 conference of the north american chapter of the association for*

*computational linguistics: Human language technologies*, 2013, pp. 746–751.

[39] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Neural Information Processing Systems (NIPS)*, 2013, pp. 1–9.

[40] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 28, no. 1, 2014.

[41] G. Ji, S. He, L. Xu, K. Liu, and J. Zhao, "Knowledge graph embedding via dynamic mapping matrix," in *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing (volume 1: Long papers)*, 2015, pp. 687–696.

[42] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, 2015.

[43] G. Ji, K. Liu, S. He, and J. Zhao, "Knowledge graph completion with adaptive sparse transfer matrix," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, 2016.

[44] M. Nickel, V. Tresp, and H.-P. Kriegel, "A three-way model for collective learning on multi-relational data," in *Icml*, 2011.

[45] B. Yang, W.-t. Yih, X. He, J. Gao, and L. Deng, "Embedding entities and relations for learning and inference in knowledge bases," *arXiv preprint arXiv:1412.6575*, 2014.

[46] M. Nickel, L. Rosasco, and T. Poggio, "Holographic embeddings of knowledge graphs," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, 2016.

[47] A. Bordes, X. Glorot, J. Weston, and Y. Bengio, "A semantic matching energy function for learning with multi-relational data," *Machine Learning*, vol. 94, no. 2, pp. 233–259, 2014.

[48] R. Socher, D. Chen, C. D. Manning, and A. Ng, "Reasoning with neural tensor networks for knowledge base completion," in *Advances in neural information processing systems*. Citeseer, 2013, pp. 926–934.

[49] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and deep locally connected networks on graphs," in *2nd International Conference on Learning Representations, ICLR 2014*, 2014.

[50] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[51] ——, "Variational graph auto-encoders," *arXiv preprint arXiv:1611.07308*, 2016.

[52] W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 1025–1035.

[53] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[54] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *European semantic web conference*. Springer, 2018, pp. 593–607.

[55] C. Shang, Y. Tang, J. Huang, J. Bi, X. He, and B. Zhou, "End-to-end structure-aware convolutional networks for knowledge base completion," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 3060–3067.

[56] T. Dettmers, P. Minervini, P. Stenetorp, and S. Riedel, "Convolutional 2d knowledge graph embeddings," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[57] D. Nathani, J. Chauhan, C. Sharma, and M. Kaul, "Learning attention-based embeddings for relation prediction in knowledge graphs," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 4710–4723.

[58] S. Vashishth, S. Sanyal, V. Nitin, and P. Talukdar, "Composition-based multi-relational graph convolutional networks," in *International Conference on Learning Representations*, 2019.

[59] Y. Levine, B. Lenz, O. Dagan, O. Ram, D. Padnos, O. Sharir, S. Shalev-Shwartz, A. Shashua, and Y. Shoham, "Sensebert: Driving some sense into bert," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 4656–4667.

[60] P. Ke, H. Ji, S. Liu, X. Zhu, and M. Huang, "SentiLARE: Sentiment-aware language representation learning with linguistic knowledge," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 6975–6988.

[61] W. Xiong, J. Du, W. Y. Wang, and V. Stoyanov, "Pretrained encyclopedia: Weakly supervised knowledge-pretrained language model," in *International Conference on Learning Representations*, 2019.

[62] A. Lauscher, I. Vulić, E. M. Ponti, A. Korhonen, and G. Glavaš, "Specializing unsupervised pretraining models for word-level semantic similarity," in *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 1371–1383.

[63] T. Shen, Y. Mao, P. He, G. Long, A. Trischler, and W. Chen, "Exploiting structured knowledge in text via graph-guided representation learning," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 8980–8994.

[64] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman, "Glue: A multi-task benchmark and analysis platform for natural language understanding," in *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, 2018, pp. 353–355.

[65] M. T. Pilehvar and J. Camacho-Collados, "Wic: the word-in-context dataset for evaluating context-sensitive meaning representations," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 1267–1273.

[66] C. Sun, L. Huang, and X. Qiu, "Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 380–385.

[67] H. Xu, B. Liu, L. Shu, and P. Yu, "Bert post-training for review reading comprehension and aspect-based sentiment analysis," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, 2019.

[68] Z. Li, X. Ding, and T. Liu, "Story ending prediction by transferable bert," *arXiv preprint arXiv:1905.07504*, 2019.

[69] D. Yin, T. Meng, and K.-W. Chang, "Sentibert: A transferable transformer-based architecture for compositional sentiment semantics," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 3695–3706.

[70] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 conference on empirical methods in natural language processing*, 2013, pp. 1631–1642.

[71] B. Pang and L. Lee, "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales," in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, 2005, pp. 115–124.

[72] A. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, 2011, pp. 142–150.

[73] X. Zhang, J. Zhao, and Y. Lecun, "Character-level convolutional networks for text classification," *Advances in Neural Information Processing Systems*, vol. 2015, pp. 649–657, 2015.

[74] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar, "Semeval-2014 task 4: Aspect based sentiment analysis," in *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, 2014, pp. 27–35.

[75] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. Al-Smadi, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq *et al.*, "Semeval-2016 task 5: Aspect based sentiment analysis," in *International workshop on semantic evaluation*, 2016, pp. 19–30.

[76] J. Berant, A. Chou, R. Frostig, and P. Liang, "Semantic parsing on freebase from question-answer pairs," in *Proceedings of the 2013 conference on empirical methods in natural language processing*, 2013, pp. 1533–1544.

[77] M. Dunn, L. Sagun, M. Higgins, V. U. Guney, V. Cirik, and K. Cho, "Searchqa: A new q&a dataset augmented with context from a search engine," *arXiv preprint arXiv:1704.05179*, 2017.

[78] B. Dhingra, K. Mazaitis, and W. W. Cohen, "Quasar: Datasets for question answering by search and reading," *arXiv preprint arXiv:1707.03904*, 2017.

[79] M. Joshi, E. Choi, D. S. Weld, and L. Zettlemoyer, "Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 1601–1611.

[80] X. Ling, S. Singh, and D. S. Weld, "Design challenges for entity linking," *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 315–328, 2015.

[81] A. Talmor, J. Herzig, N. Lourie, and J. Berant, "Commonsenseqa: A question answering challenge targeting commonsense knowledge," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4149–4158.

[82] M. Sap, H. Rashkin, D. Chen, R. Le Bras, and Y. Choi, "Social iqa: Commonsense reasoning about social interactions," 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)*, 2019, pp. 4453–4463.

[83] T. Dettmers, P. Minervini, P. Stenetorp, and S. Riedel, "Convolutional 2d knowledge graph embeddings," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[84] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Neural Information Processing Systems (NIPS)*, 2013, pp. 1–9.

[85] Z. Zhang, X. Han, Z. Liu, X. Jiang, M. Sun, and Q. Liu, "Ernie: Enhanced language representation with informative entities," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 1441–1451.

[86] M. E. Peters, M. Neumann, R. Logan, R. Schwartz, V. Joshi, S. Singh, and N. A. Smith, "Knowledge enhanced contextual word 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)*, 2019, pp. 43–54.

[87] I. Balazevic, C. Allen, and T. Hospedales, "Tucker: Tensor factorization for knowledge graph completion," 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)*, 2019, pp. 5188–5197.

[88] B. He, D. Zhou, J. Xiao, X. Jiang, Q. Liu, N. J. Yuan, and T. Xu, "Integrating graph contextualized knowledge into pre-trained language models," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, 2020, pp. 2281–2290.

[89] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, "Biobert: a pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, 2020.

[90] I. Beltagy, A. Cohan, and K. Lo, "Scibert: Pretrained contextualized embeddings for scientific text," *arXiv preprint arXiv:1903.10676*, vol. 1, no. 1.3, p. 8, 2019.

[91] À. Bravo, J. Piñero, N. Queralt-Rosinach, M. Rautschka, and L. I. Furlong, "Extraction of relations between genes and diseases from text and large-scale data analysis: implications for translational research," *BMC bioinformatics*, vol. 16, no. 1, pp. 1–17, 2015.

[92] E. M. Van Mulligen, A. Fourrier-Reglat, D. Gurwitz, M. Molokhia, A. Nieto, G. Trifiro, J. A. Kors, and L. I. Furlong, "The eu-adr corpus: annotated drugs, diseases, targets, and their relationships," *Journal of biomedical informatics*, vol. 45, no. 5, pp. 879–884, 2012.

[93] B. Bhasuran and J. Natarajan, "Automatic extraction of gene-disease associations from literature using joint ensemble learning," *PloS one*, vol. 13, no. 7, p. e0200699, 2018.

[94] B. He, Y. Guan, and R. Dai, "Classifying medical relations in clinical text via convolutional neural networks," *Artificial intelligence in medicine*, vol. 93, pp. 43–49, 2019.

[95] A. Yang, Q. Wang, J. Liu, K. Liu, Y. Lyu, H. Wu, Q. She, and S. Li, "Enhancing pre-trained language representations with rich knowledge for machine reading comprehension," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 2346–2357.

[96] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. Hruschka, and T. Mitchell, "Toward an architecture for never-ending language learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 24, no. 1, 2010.

[97] S. Zhang, X. Liu, J. Liu, J. Gao, K. Duh, and B. Van Durme, "Record: Bridging the gap between human and machine commonsense reading comprehension," *arXiv preprint arXiv:1810.12885*, 2018.

[98] X. Liu, Y. Shen, K. Duh, and J. Gao, "Stochastic answer networks for machine reading comprehension," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 1694–1704.

[99] C. Clark and M. Gardner, "Simple and effective multi-paragraph reading comprehension," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 845–855.

[100] A. W. Yu, D. Dohan, M.-T. Luong, R. Zhao, K. Chen, M. Norouzi, and Q. V. Le, "Qanet: Combining local convolution with global self-attention for reading comprehension," in *International Conference on Learning Representations*, 2018.

[101] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "Squad: 100,000+ questions for machine comprehension of text," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 2383–2392.

[102] R. Logan, N. F. Liu, M. E. Peters, M. Gardner, and S. Singh, "Barack's wife hillary: Using knowledge graphs for fact-aware language modeling," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 5962–5971.

[103] S. Merity, N. S. Keskar, and R. Socher, "Regularizing and optimizing lstm language models," in *International Conference on Learning Representations*, 2018.

[104] Y. Ji, C. Tan, S. Martschat, Y. Choi, and N. A. Smith, "Dynamic entity representations in neural language models," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 1830–1839.

[105] J. Guan, F. Huang, Z. Zhao, X. Zhu, and M. Huang, "A knowledge-enhanced pretraining model for commonsense story generation," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 93–108, 2020.

[106] O. Levy, M. Seo, E. Choi, and L. Zettlemoyer, "Zero-shot relation extraction via reading comprehension," in *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, 2017, pp. 333–342.

[107] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin, "Convolutional sequence to sequence learning," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1243–1252.

[108] A. Fan, M. Lewis, and Y. Dauphin, "Hierarchical neural story generation," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 889–898.

[109] L. Yao, N. Peng, R. Weischedel, K. Knight, D. Zhao, and R. Yan, "Plan-and-write: Towards better automatic storytelling," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 7378–7385.

[110] S. Rose, D. Engel, N. Cramer, and W. Cowley, "Automatic keyword extraction from individual documents," *Text mining: applications and theory*, vol. 1, pp. 1–20, 2010.

[111] J. Xu, X. Ren, Y. Zhang, Q. Zeng, X. Cai, and X. Sun, "A skeleton-based model for promoting coherence among sentences in narrative story generation," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 4306–4315.

[112] A. Fan, M. Lewis, and Y. Dauphin, "Strategies for structuring story generation," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 2650–2660.

[113] W. Liu, P. Zhou, Z. Zhao, Z. Wang, Q. Ju, H. Deng, and P. Wang, "K-bert: Enabling language representation with knowledge graph," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 03, 2020, pp. 2901–2908.

[114] B. Xu, Y. Xu, J. Liang, C. Xie, B. Liang, W. Cui, and Y. Xiao, "Cn-dbpedia: A never-ending chinese knowledge extraction system," in *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*. Springer, 2017, pp. 428–438.

[115] Z. Dong and Q. Dong, "Hownet and the computation of meaning: (with cd-rom)."

[116] T. Sun, Y. Shao, X. Qiu, Q. Guo, Y. Hu, X.-J. Huang, and Z. Zhang, "Colake: Contextualized language and knowledge embedding,"

in *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 3660–3670.

[117] X. Wang, T. Gao, Z. Zhu, Z. Zhang, Z. Liu, J. Li, and J. Tang, "Kepler: A unified model for knowledge embedding and pre-trained language representation," *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 176–194, 2021.

[118] P. Verga, H. Sun, L. B. Soares, and W. Cohen, "Adaptable and interpretable neural memoryover symbolic knowledge," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2021, pp. 3678–3691.

[119] K. Jiang, D. Wu, and H. Jiang, "Freebaseqa: a new factoid qa data set matching trivia-style question-answer pairs with freebase," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 318–323.

[120] W.-t. Yih, M.-W. Chang, X. He, and J. Gao, "Semantic parsing via staged query graph generation: Question answering with knowledge base," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2015, pp. 1321–1331.

[121] H. Sun, A. O. Arnold, T. Bedrax-Weiss, F. Pereira, and W. W. Cohen, "Guessing what's plausible but remembering what's true: Accurate neural reasoning for question-answering," *arXiv preprint arXiv:2004.03658*, 2020.

[122] T. Févry, L. B. Soares, N. FitzGerald, E. Choi, and T. Kwiatkowski, "Entities as experts: Sparse memory access with entity supervision," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 4937–4951.

[123] R. Wang, D. Tang, N. Duan, Z. Wei, X. Huang, C. Cao, D. Jiang, M. Zhou *et al.*, "K-adapter: Infusing knowledge into pre-trained models with adapters," *arXiv preprint arXiv:2002.01808*, 2020.

[124] L. Huang, R. Le Bras, C. Bhagavatula, and Y. Choi, "Cosmos qa: Machine reading comprehension with contextual commonsense reasoning," 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)*, 2019, pp. 2391–2401.

[125] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M.-W. Chang, "Realm: Retrieval-augmented language model pre-training," *arXiv preprint arXiv:2002.08909*, 2020.

[126] K. Lee, M.-W. Chang, and K. Toutanova, "Latent retrieval for weakly supervised open domain question answering," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 6086–6096.

[127] A. Shrivastava and P. Li, "Asymmetric lsh (alsh) for sublinear time maximum inner product search (mips)," *Advances in Neural Information Processing Systems*, vol. 3, no. January, pp. 2321–2329, 2014.

[128] T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, J. Devlin, K. Lee *et al.*, "Natural questions: a benchmark for question answering research," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 453–466, 2019.

[129] D. Chen, A. Fisch, J. Weston, and A. Bordes, "Reading wikipedia to answer open-domain questions," in *55th Annual Meeting of the Association for Computational Linguistics, ACL 2017*. Association for Computational Linguistics (ACL), 2017, pp. 1870–1879.

[130] S. Min, D. Chen, H. Hajishirzi, and L. Zettlemoyer, "A discrete hard em approach for weakly supervised question answering," 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)*, 2019, pp. 2844–2857.

[131] S. Min, D. Chen, L. Zettlemoyer, and H. Hajishirzi, "Knowledge guided text retrieval and reading for open domain question answering," *arXiv preprint arXiv:1911.03868*, 2019.