Computer Modeling in Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2023.031513

REVIEW

A Survey of Knowledge Graph Construction Using Machine Learning

Zhigang Zhao[1], Xiong Luo[1,2,3,*], Maojian Chen[1,2,3] and Ling Ma[1]

[1]School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing, 100083, China

[2]Shunde Innovation School, University of Science and Technology Beijing, Foshan, 528399, China

[3]Beijing Key Laboratory of Knowledge Engineering for Materials Science, Beijing, 100083, China

*Corresponding Author: Xiong Luo. Email: xluo@ustb.edu.cn

Received: 25 June 2023 Accepted: 13 September 2023 Published: 30 December 2023

ABSTRACT

Knowledge graph (KG) serves as a specialized semantic network that encapsulates intricate relationships among real-world entities within a structured framework. This framework facilitates a transformation in information retrieval, transitioning it from mere string matching to far more sophisticated entity matching. In this transformative process, the advancement of artificial intelligence and intelligent information services is invigorated. Meanwhile, the role of machine learning method in the construction of KG is important, and these techniques have already achieved initial success. This article embarks on a comprehensive journey through the last strides in the field of KG via machine learning. With a profound amalgamation of cutting-edge research in machine learning, this article undertakes a systematical exploration of KG construction methods in three distinct phases: entity learning, ontology learning, and knowledge reasoning. Especially, a meticulous dissection of machine learning-driven algorithms is conducted, spotlighting their contributions to critical facets such as entity extraction, relation extraction, entity linking, and link prediction. Moreover, this article also provides an analysis of the unresolved challenges and emerging trajectories that beckon within the expansive application of machine learning-fueled, large-scale KG construction.

KEYWORDS

Knowledge graph (KG); semantic network; relation extraction; entity linking; knowledge reasoning

1 Introduction

The continuous development of information technologies brings significant convenience to human life, paralleled by an exponential surge in information proliferation. Massive amounts of data are collected and studied in many fields such as social network, biomedical engineering, security science, and many others. Under this background, search engine has become an indispensable instrument, facilitating people's quest for knowledge and information online. Traditionally, a search engine has a user input a query term, whereupon it furnishes hyperlinks directing to the most relevant web pages corresponding to the provided keyword [1].

In May 2012, the emergence of knowledge graph (KG) brought a novel paradigm for enhancing search engines. Within this framework, user search results transcend the realm of single web page links,

encompassing instead a tapestry of structured entity information closely related to the search query. This transformative approach even delves into the realm of potential hidden knowledge within the KG. The intelligent optimization of search answers through KG can effectively improve the functions of future search engines in three aspects: refining responses, nurturing interactive dialogues, and bolstering predictive capabilities [2]. This multifaceted augmentation leads into an era of heightened search engine functionality. Furthermore, the scope of KG's influence extends considerably into domains beyond search, encompassing intelligent question answering, knowledge engineering, data mining, and digital library.

Recent years have witnessed a remarkable surge of interest from various disciplines engineering and science. As depicted in Fig. 1 and supported by data from the Web of Science, the number of published papers with "Knowledge Graph" in their title has exhibited a steady rise up until the end of 2022. Additionally, Figs. 2–4 provide insight into the volume of published papers across diverse research areas, publication resources, and institutions. In the field of academic research, Computer Science has emerged as a primary hub for the propagation of KG, closely followed by Mathematics and Engineering. This observation highlights the substantial contribution and enthusiasm originating from the Computer Science towards the exploration, development, and advancement of KG-related topics. Notably, the significant research output in Mathematics and Engineering emphasizes the interdisciplinary essence of KGs, signifying their influence across domain that extend beyond the realm of computer. Shifting the focus to the publication platforms, it is evident that *Lecture Notes in Computer Science* has emerged as the leading platform for KG-centric research, closely followed by *Lecture Notes in Artificial Intelligence* and *IEEE Access*. This indicates that *Lecture Notes in Computer Science* has been the preferred choice for researchers to share their findings and advancements in the field of KGs. Lastly, the Chinese Academy of Sciences, University of Chinese Academy of Science, and Rluk Research Libraries UK have emerged as the leading contributors in terms of publishing papers related to KG. These institutions have demonstrated a robust presence and active participation in KG research, highlighting their expertise and dedicated contributions to the advancement of this field. Their significant published works attest to the valuable role played by these institutions in the exploration and evolution of KG-related topics.

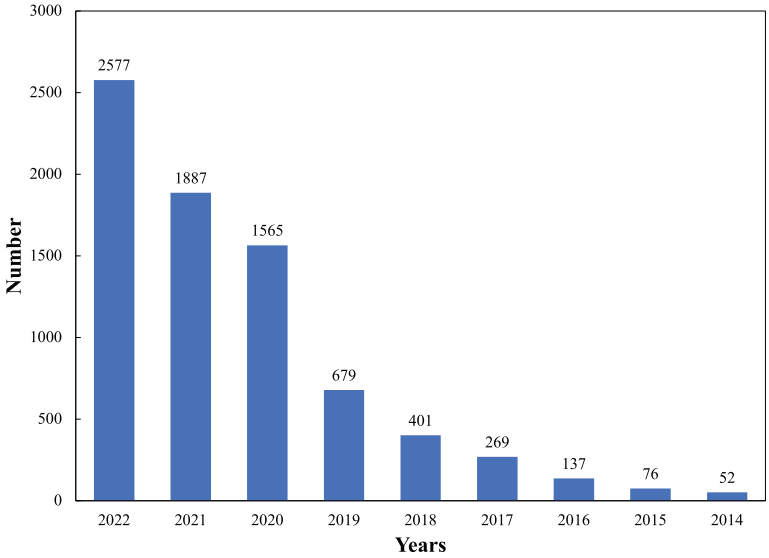

Figure 1: Number of published papers with "Knowledge Graph" in the title in recent years

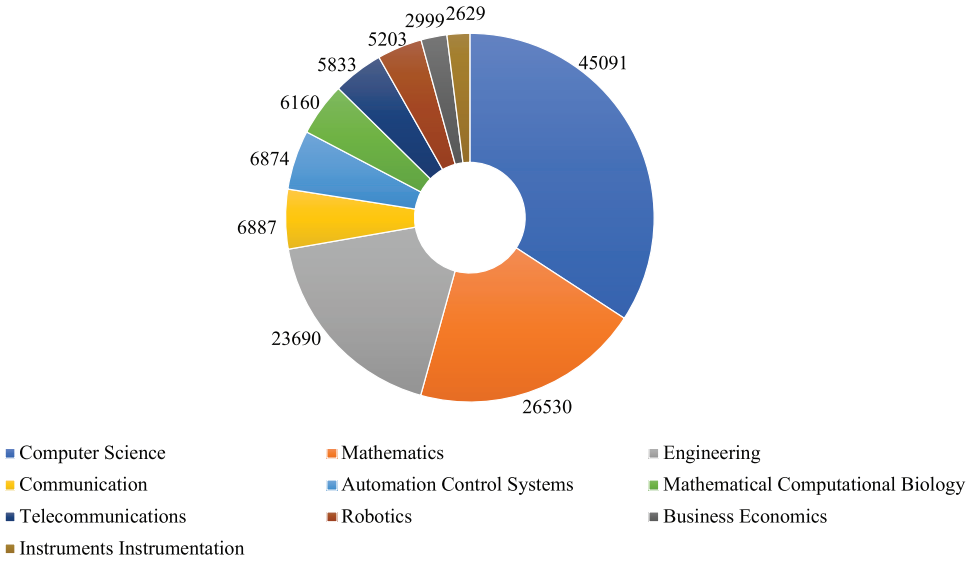

Figure 2: The top 10 research areas ranked by the number of publication related to KG

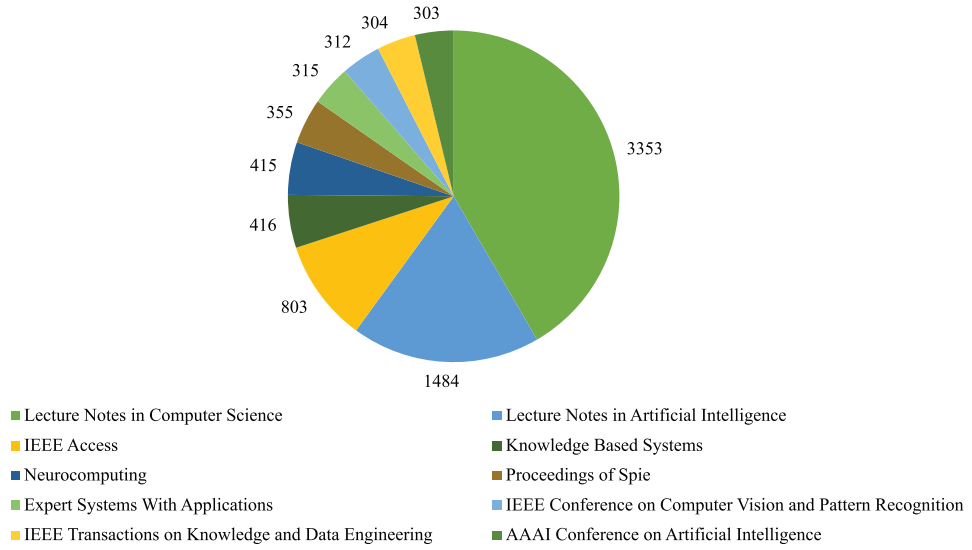

Figure 3: The top 10 publication resources ranked by the number of publication in KG

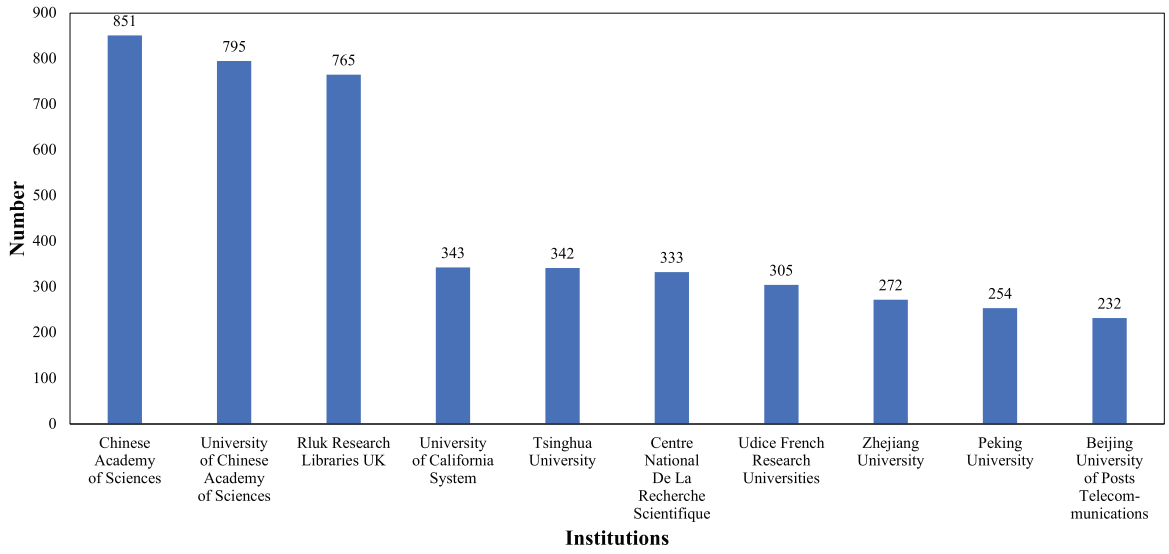

Figure 4: The top 10 institutions ranked by the number of published papers related to KG

Generally speaking, the establishment and utilization of large-scale KG necessitate the synergy of diverse intelligent information processing technologies. In recent years, machine learning methods have as a pivotal force within the KG construction. Therefore, this article conducts a survey in relation to this field. We first introduce the development history, fundamental principles and technical framework of KG. Second, we summarize the machine learning-based key technologies integral to the actualization, dissected across three pivotal dimensions. Finally, we expound upon the current challenges and future forthcoming trends poised to guide the construction of large-scale KG. The main contributions are as follows:

- This article offers an extensive and up-to-date review of existing research and literature in the field of KG construction, specifically focusing on methodologies driven by machine learning techniques. Meanwhile, it provides a structured categorization and classification of diverse machine learning-driven approaches utilized for constructing KGs. This categorization could help readers understand the landscape and taxonomy of methods in this field.
- This article delves into the various machine learning methodologies employed in the construction of KGs. Meanwhile, we conduct the comparative evaluation and analysis of different machine learning techniques, showcasing their respective performance, scalability, and suitability under various conditions.
- This article identifies and discusses challenges and open research questions within the domain of KG construction using machine learning, highlighting potential avenues for further exploration and innovation.

The organization of this article is arranged as follows. In Section 2, we present some fundamental concepts and traditional technical architecture of KG. Section 3 focuses on a comprehensive exploration of KG design propelled by the prowess of machine learning methods, meticulously partitioned into three parts, i.e., entity learning, ontology learning, and knowledge reasoning. In Section 4, we discuss the prospective research directions and challenges of large-scale KG construction technologies. Through machine learning methods, we dissect this discussion into three distinct segments, relation extraction, link prediction, and construction of industrial KG. Section 5 provides a conclusion and reflections.

2 Knowledge Graph

In this section, we simply introduce the essence and the foundational framework of KG.

2.1 The Development of Knowledge Graph

With the development of the Internet, Web technology has gone through the "Web 1.0" era characterized by the web of documents and the "Web 2.0" era characterized by the web of data. Today, the trajectory points towards the "Web 3.0" era characterized by the web of knowledge [3] and even anticipates the "Web 4.0" era defined by the Metaverse paradigm [4]. Driven by the continuous growth of user-generated content and open-linked data on the Internet, the quest for knowledge interconnected aligning with the ever-evolving network information resources becomes imperative. This quest takes a fresh perspective in accordance with the principles of knowledge organization in the big data environment, aiming to reveal deeper cognitive insights [5]. In the midst of this dynamic context, Google introduced KG in May 2012. Its goal is to enhance search outcomes, describe the various entities and concepts inherent to the real world, and illuminate their relationships. By these merits, KG emerges as a substantial stride forward from prevailing semantic web technologies. Illustrated in Fig. 5 are pivotal milestones making the history of KG across different years. For instance, conception of the semantic network as a vehicle for knowledge representation was proposed in 1960. Furthermore, the philosophical concept of "ontology" was integrated into the KG in 1980, facilitating the structured and formalized description of knowledge.

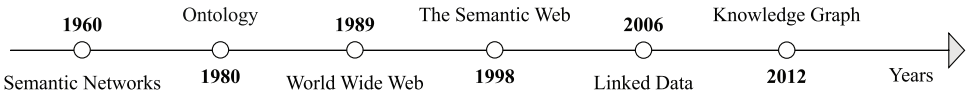

Figure 5: Milestones in the development of KG

The origin of designing KG comes from a series of practical applications, spanning fields such as semantic search, machine question answering, information retrieval, online learning and others. With the exploration of KG advances, various structured KGs have been developed by both academic researchers and industry practitioners. Currently, a tableau of prominent and expansive large-scale open knowledge bases associated with KG exists globally, as enumerated in Table 1. Here, large-scale knowledge bases like Freebase [6], DBPedia [7], and Wikidata [8] take center stage using Wikipedia as a foundational source. Notably, Freebase differentiates itself by its user-generated content, open accessibility, and structured data, which supports all its entries.

Table 1: The sizes of some prominent large-scale knowledge bases

Knowledge graph	Number of entities	Number of relation types	Number of facts
Freebase [6]	40 M	35000	637 M
DBpedia [7]	5 M	1367	538 M
Wikidata [8]	18 M	1632	66 M
YAGO2 [9]	10 M	114	447 M
Google KG [7]	570 M	35000	18000 M

In recent years, an array of research results on the Chinese KG has grown vigorously. For example, Sogou established "Knowledge Cube", marking the inception of the foremost knowledge base search

product in the domestic search engine industry. Through effectively integrating fragmented Internet knowledge, Baidu founded "Baidu Zhixin" and brought forth a next-generation search engine product. Contributions extend to academia as well, Tsinghua University built "XLore", a pioneering large-scale Chinese-English cross-language KG. The Institute of Computing Technology of Chinese Academy of Sciences established a prototype system termed "People Cube, Work Cube, Knowledge Cube" based on an open knowledge network OpenKN. Shanghai Jiao Tong University designed 'Zhishi.me', a dedicated research platform for Chinese KG. Additionally, the GDM Lab at Fudan University launched the Chinese KG project. Generally, these products and projects have given rise to expansive knowledge bases spanning diverse fields, providing users with intelligent search and question-and-answer services.

2.2 The Definition of Knowledge Graph

KG is a special semantic network composed of nodes and directed edges, and it is also known as a heterogeneous information network or semantic knowledge base. In the KG, each node represents an entity in the real world, while directed edges interlink these nodes to denote the intricate relationships between those entities. Facts are generally represented in the form of triples (subject, predicate, object) (SPO), where subject and object signify entities, and predicate represents the relation between them [10]. For example, the textual data "Chao Deng is an actor who played the character Tailang Xu in the comedy movie Duckweed" can be expressed via the following set of SPO triples exemplified in Table 2. The transformed version of Table 2 into a KG is described in Fig. 6. This KG encapsulates the interrelationships between various entities, allowing for a structured representation of the given information.

Table 2: An example of SPO triples extracted from text data

Subject	Predicate	Object
Chao Deng	Occupation	Actor
Chao Deng	Starred in	Duckweed
Chao Deng	Played	Tailang Xu
Tailang Xu	Character in	Duckweed
Duckweed	Genre	Comedy movie

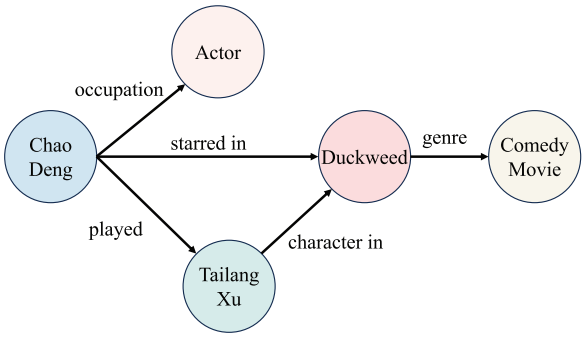

Figure 6: The updated version of Table 2 after translating it into a KG

2.3 The Technical Architecture of Knowledge Graph

The structure of KG includes two aspects: a logical structure and a technical structure. The former includes the data layer and the pattern layer, while the latter refers to the technological process involved in KG construction. This involves a sequence of stages, including data acquisition, entity learning, ontology learning, knowledge reasoning, and knowledge update.

The data layer, alternatively known as the entity layer, functions as a repository for knowledge housing information in the form of facts. These facts are succinctly conveyed through triples as the basic expression of facts, where a graph database is chosen as a storage medium.

Above the data layer, the pattern layer, commonly referred to as the ontology layer, represents and stores refined concepts and knowledge. The pattern layer leverages ontology constructs, effectively serving as an embodiment of the KG. It plays an important role in defining and organizing entities, properties, classes, and relationships within a KG. Typically, ontology is represented using formal languages like the Web Ontology Language (OWL) [11], which is grounded in description logic. OWL's expressive capabilities empower ontologies to accurately define semantic relationships among entities, properties, classes, and relationships. Moreover, ontology-based reasoning facilitates the identification of missing information in the KG, the discovery of hidden relationships between entities, and the ability to address intricate semantic queries. Furthermore, ontology enables seamless interaction and knowledge sharing among different KGs. This fosters the construction of larger, more comprehensive KG, and promotes the reuse of knowledge, thereby augmenting the overall value of knowledge graph. Generally, the axioms, rules, and constraints in the ontology base are used to standardize the entities, the types and the attributes of entities, and the relationship between the entities, so that the KG has a strong structure and less redundancy [12].

The technical architecture of KG construction can be classified into two principal paradigms: top-down and bottom-up. In the former, the pattern layer is first defined, and the construction begins from the top-level concept. It then proceeds to progressively refine, layer, and generate instances downward. The process is depicted in Fig. 7. Conversely, the latter starts from the underlying entities, extracts entities, and gradually abstracts them upwards to form upper-level concepts and knowledge. This architectural construct is illustrated in Fig. 8. It starts from the original semi-structured data or unstructured data and adopts a series of technologies to extract knowledge. Then, it integrates with the structural data, with the ontology layer contributing to the enrichment of upper-level concepts. Finally, a complete KG is generated. Furthermore, the comprehensive KG is continuously updated and augmented. New knowledge is continuously extracted to promote the KG refinement. Within this framework, each iteration generally includes data acquisition, entity learning, ontology learning, knowledge reasoning, and knowledge updating [13].

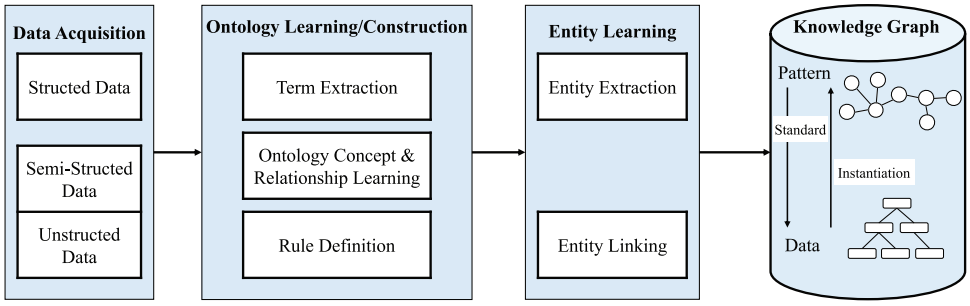

Figure 7: The technical architecture of the top-down KG

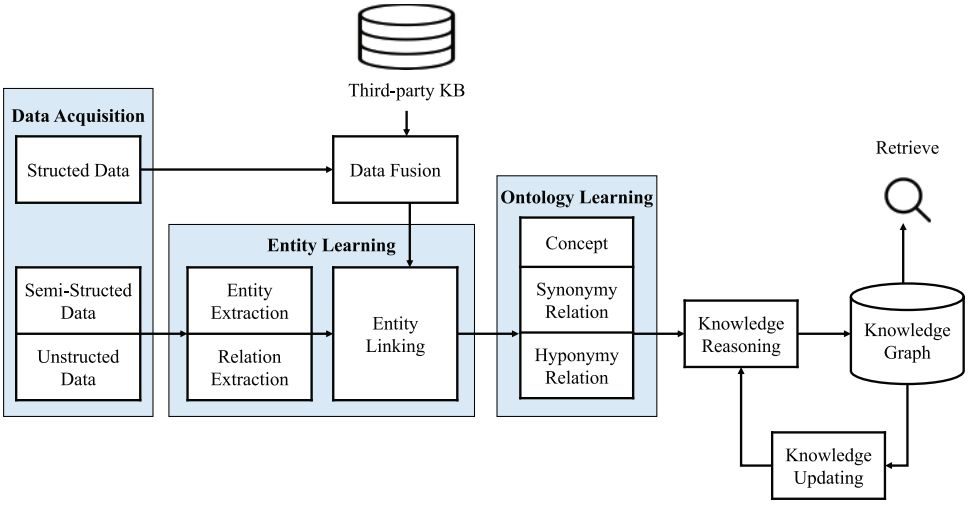

Figure 8: The technical architecture of bottom-up KG

3 Knowledge Graph Construction Using Machine Learning

A typical machine learning approach is usually operated on a structured data matrix, where each row in the matrix corresponds to an object characterized by an attribute eigenvector. The main task of machine learning is to achieve the mapping from these eigenvectors to various forms of output through learning. Additionally, unsupervised learning can facilitate clustering and factor analysis. Then, according to the bottom-up KG construction process described in Fig. 8, we summarize the realization of KG driven by machine learning from the following three parts, each bearing its unique set of implementation techniques and challenges.

3.1 Entity Learning

Entity learning refers to the intricate construction process of entity layer within the KG. From bottom to top, it includes three modules: entity extraction, relationship extraction, and entity linking.

3.1.1 Entity Extraction

Entity extraction stands as the initial and pivotal step in the knowledge extraction, involving the automatic identification of named entities from the original corpus. This foundational process relies on the automatic detection and categorization of named entities within a given corpus. Fig. 9 illustrates various categories to which these named entities can be ascribed, such as Person, Country, City, and more. This process enables the identification of important entities, laying the foundation for subsequent knowledge extraction and analysis tasks.

Generally, there are three typical methods in this field, and they are the rule-based, the traditional machine learning-based, and deep learning-based extraction methods. Fig. 10 displays the classification results obtained from these methods, showcasing the effectiveness and performance of each strategy in identifying and categorizing named entities. Here, rule extraction is an early pattern implemented by manually designing rules primarily toward proper nouns within a specific domain's text. It is based on painstakingly handcrafted patterns, necessitating a lot of human efforts, culminating in limited extraction capacity and constrained scalability. Meanwhile, the related extraction methods based on machine learning models have witnessed remarkable progress. These methods integrate

machine learning algorithms into entity extraction to achieve automatic or semi-automatic entity identification. Finally, with the development of artificial neural networks, some deep learning-based methods have been proposed to attain heightened proficiency in entity extraction tasks with reduced human intervention. This progression signifies a remarkable leap forward in the field.

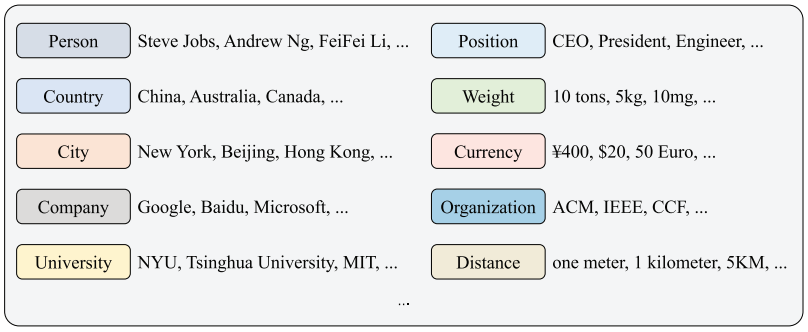

Figure 9: Examples of different types of entities

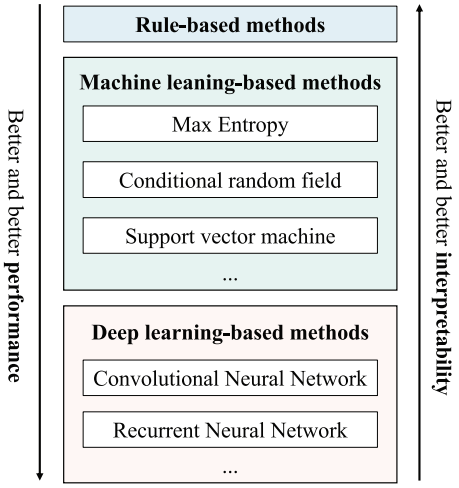

Figure 10: Classification results of entity extraction methods

For the traditional machine learning-based entity extraction, the inception involved the utilization of supervised machine learning algorithms. For example, the decision trees and the conditional random field (CRF) model [14] were employed to realize entity recognition for Telugu-English code-mixed social media data. In addition, Sykes et al. [15] conducted a comprehensive comparison between rule-based and machine learning-based extraction methods, effectively emphasizing the latter's superior adaptability and versatility.

Furthermore, with the progressive evolution of web technologies, the combination of open-linked data with machine learning algorithms frequently yields more compelling results. Within this framework, the basic idea revolves around employing machine learning to extract entities with similar contextual features from the web page, subsequently achieving entity classification or clustering. For example, Whitelaw et al. [16] proposed an iterative approach for expanding the entity corpus in a network environment. This approach depended on the construction of feature models grounded in known entities, enabling the processing of massive datasets. With this mechanism, they effectively

modeled new entities to achieve continuous iterative expansion of the entity. Furthermore, employing unsupervised learning algorithms, Jain and Pennacchiotti [17] successfully extracted newly emerging named entities from the server logs of search engines. Then, this method found practical application in search engine technology, allowing for automatic information completion based on user-input keywords. Additionally, while constructing KG, an application of intelligent corpus annotation for entity extraction was presented [18].

Deep learning-based methods offer the advantage of automated text feature selection for entity extraction, which reduces incompleteness and manual work. They have shown promising results in this area. One such method employs convolutional neural networks (CNN) to automatically learn features from the input text data. For example, a deep learning-based method [19–21] was proposed for entity extraction that utilized a CNN to learn contextual features from the input text data. Those models were trained on a large dataset of annotated text datasets and exhibited satisfactory performance on several benchmark datasets. Similarly, Cho et al. [22] presented a deep learning-based strategy for entity recognition in biomedical texts, combining a CNN with a long short-term memory (LSTM) network. Their model acquired complex features from the input text data, attaining remarkable accuracy on a challenging biomedical entity recognition task. Overall, deep learning-based methods have the potential to advance entity extraction, thereby constructing more precise and comprehensive KG. Hence, this enhancement leads to more powerful applications in many fields, such as natural language processing (NLP) and information retrieval. Notably, Table 3 showcases the top 10 entity extraction models from 2019 to 2022, ranked by F1-score on various open-source datasets (CoNLL 2003 [23], ACE 2005 [24], and Ontonotes 2005 [25]) This information is sourced from the Papers with Code website (https://paperswithcode.com).

Table 3: The top 10 entity extraction models from 2019 to 2022

Datasets	Model	F1-score (%)	Year	Platform
CoNLL 2003	ACE+document-context [26]	94.60	2021	Github
	Co-regularized LUKE [27]	94.22	2021	Github
	ASP+T5-3B [28]	94.10	2022	Github
	FLERT XLM-R [29]	94.09	2020	Github/HuggingFace
	PL-Marker [30]	94.00	2022	Github
	LUKE [31]	93.91	2020	Github/HuggingFace
	CL-KL [32]	93.85	2021	Github
	XLNet-GCN [33]	93.82	2021	Github
	ASP+flan-T5-large [28]	93.80	2022	Github
	InferNER [34]	93.76	2021	–
ACE 2005	PURE [35]	90.90	2021	Github
	PromptNER[RoBERTa-large] [36]	88.26	2023	Github
	PIQN [37]	87.42	2022	Github
	PromptNER[BERT-large] [36]	87.21	2023	Github
	DiffusionNER [38]	86.93	2023	Github
	BERT-MRC [39]	86.88	2020	Github
	Locate and Label [40]	86.67	2021	Github
	BoningKnife [41]	85.46	2021	–

(Continued)

Table 3 (continued)

Datasets	Model	F1-score (%)	Year	Platform
	Biaffine-NER [42]	85.40	2020	Github
	Second-best learning and decoding [43]	84.34	2020	Github
Ontonotes 2005	BERT-MRC+DSC [44]	92.07	2019	Github
	PL-Marker [30]	91.90	2022	Github
	Baseline+BS [45]	91.74	2022	Github
	Biaffine-NER [42]	91.30	2020	Github
	BERT-MRC [39]	91.11	2020	Github
	PIQN [37]	90.96	2022	Github
	HGN [46]	90.92	2022	Github
	Syn-LSTM+BERT [47]	90.85	2021	Github
	DiffusionNER [38]	90.66	2023	Github
	W2NER [48]	90.50	2022	Github

Notes: For more detailed and up-to-date information about the models than what is presented in the article, readers can refer to the following links: https://paperswithcode.com/sota/named-entity-recognition-ner-on-conll-2003; https://paperswithcode.com/sota/named-entity-recognition-on-ace-2005; https://paperswithcode.com/sota/named-entity-recognition-ner-on-ontonotes-v5.

For the CoNLL 2003 dataset, several remarkable models have garnered high F1-score, highlighting their prowess in entity extraction. The Automated Concatenation of Embeddings (ACE) combined with document-context [26] stands out at an impressive 94.60%. Close on its heels, the Co-regularized Language Understanding with Knowledge-based Embeddings (LUKE) [27] achieves a commendable 94.22%, while the Autoregressive Structured Prediction (ASP) fused with Text-to-Text Transfer Transformer (T5)-3B [28] achieves a noteworthy 94.10% (the "3B" represents 3 billion parameters). These models have demonstrated outstanding performance in extracting entities from the CoNLL 2003 dataset.

In the ACE 2005 dataset, the Princeton University Relation Extraction system (PURE) [35] emerges triumphant with an F1-score of 90.90%. Not far behind, PromptNER [36] has a second position with 88.26%, followed by Parallel Instance Query Network (PIQN) [37] with 87.42%. These models have shown strong performance in entity extraction from the ACE 2005 dataset.

For the Ontonotes 2005 dataset, the Bidirectional Encoder Representations from Transformers (BERT)-Machine Reading Comprehension (MRC)+dice coefficient (DSC) [44] attains the pinnacle with the highest F1-score of 92.07%. Packed Levitated (PL)-Marker [30] follows closely behind with 91.90%, closely trailed by Baseline+Boundary Smoothing (BS) [45] achieving 91.74%. These models have demonstrated their efficacy in entity extraction from the Ontonotes 2005 dataset.

These state-of-the-art models serve as prime examples of the evolutions made in entity extraction techniques, leveraging various approaches including deep learning, Prompt, and co-regularization methods. With remarkable F1-score on their respective datasets, these models indicate their effectiveness in extracting entities from text.

3.1.2 Relation Extraction

Relation extraction is an important subtask of knowledge extraction. In the past, this task entailed manual rule construction, followed by pattern-matching techniques to extract corresponding relation

instances from text. However, the advent of machine learning has revolutionized relation extraction methods. It leverages lexical and syntactic attributes for model training, effectively transmuting relation extraction challenges into classification or clustering problems. According to the extent of human involvement and dependence on labeled corpus, machine learning-based relation extraction approaches can be divided into the supervised learning-based relation extraction, the semi-supervised learning-based relation extraction, and the unsupervised learning-based relation extraction.

(1) Relation Extraction with Supervised Learning Method

Supervised learning-based relation extraction is an automatic mode on the basis of meticulously labeled training data. Through ongoing learning from these training samples, the classification and predictions are performed on datasets. Within this paradigm, binary relation extraction is treated as a classification problem. As shown in the following definition, the triple (e_i, r_k, e_j) indicates that there is a semantic relationship r_k between the head entity e_i and the tail entity e_j, and function $f(\cdot)$ represents the relationship classifier employed in the context:

$$f(e_i, r_k, e_j) = \begin{cases} 1 & \text{if the triple}(e_i, r_k, e_j) \text{ exists,} \\ 0 & \text{otherwise.} \end{cases}$$

Supervised learning employs two primary categories of relationship classification methods: feature vector-based and kernel function-based. In the first category, features are extracted from the training samples and represented as sequence feature vectors, using the results of part-of-speech tagging and syntax parsing. Here, the prominent methods are support vector machine (SVM) [49] and maximum entropy (ME) [50]. For example, a classifier system employing SVM integrated lexical features from polarity lexicons and lists of offensive/profane words to identify and classify offensive language in social media [49]. The second category of methods effectively avoids the challenges of dimensionality caused by nonlinear transformations. Recent years have witnessed widespread utilization of kernel function-based approaches in many fields [51]. In relation extraction using kernel functions, such as convolution kernel [52], tree kernel [53], subsequence kernel [54] and some improved kernel methods [55] play a pivotal role. The key of these methods involves projecting the implicit feature vector of a sentence into the feature space using the kernel function and calculating the inner product between these projections, so as to assess the similarity of the relationship between entities. For example, an innovative tree kernel, termed feature-enriched tree kernel (FTK) was proposed [53], while achieving a 5.4% enhancement in F-measure over the traditional convolution tree kernel.

Supervised learning-based relation extraction methods yield excellent experimental results, but their effectiveness heavily depends on the classification features provided by part-of-speech tagging and syntactic parsing. To address this problem, the use of supervised relation extraction has witnessed a surge of interest driven by the deep learning model [56]. A recurrent neural network (RNN)-based relation extraction model was proposed by Socher et al. [57]. This approach involved vectorizing each node of the syntactic tree through syntactic analysis. Guided by the syntactic structure, it iterated continuously from the lowest word vector of the tree. Finally, the vector representation of the sentence was attained and employed as the foundation for relationship classification. This method effectively utilized syntactic structure information, but overlooked the position information of words. Then, convolutional neural network (CNN) took center stage. Here, the word vector was treated as an initialization parameter, engaging in convolution training with dynamic optimization during the learning process, culminating in classification [58]. To complement the local dependencies captured by piecewise CNNs, a self-attention mechanism was proposed to capture rich contextual dependencies [59]. The experiments were performed on the NYT dataset and the experimental results demonstrated

that the model provided a new benchmark in the area under curve (AUC) metric. Expanding on the CNN-based methodology presented [57], a refined iteration emerged [60]. This advancement involved inputting both word vectors and word positional vectors, with the sentence representation being obtained through the learning of convolutional, pooling, and nonlinear layers. This method fully considered the entity location information and other related lexical features, which achieved good relation extraction results. On the standard SemEval-2018 Task 7 dataset, the CNN method achieved superior performance when compared to alternative relation extraction methods [61]. Furthermore, except for deep learning models, deep reinforcement learning has also been used in the relation extraction [62]. This approach casts the relation extraction as a two-step decision-making game, employing the Q-Learning algorithm with value function approximation to learn control policy. The experiments were conducted on the ACE 2005 corpus, and they showed that the deep reinforcement learning model achieved a state-of-the-art performance in relation extraction tasks.

(2) Relation Extraction with Semi-supervised Learning Method

Semi-supervised learning-based relation extraction aims to realize the binary relation classification with limited training samples, thus circumventing the constraints imposed by manual annotation of extensive training data. It mainly adopts the bootstrapping method and some other methods.

The idea of bootstrapping is to artificially construct a small set of initial relation instances as a seed. This seed set serves as the foundation for model training, and through iterative expansion, it gradually augments to encompass a more extensive collection of relation instances, and finally completes the relation extraction task. Here, the entity alignment technique was improved to reduce the data noise [63]. However, this method operates under the assumption that a single entity pair corresponds to just one relationship. To address this limitation, a multi-instance multi-label (MIML) method was proposed to model the relationship extraction. This methodology describes the situation where an entity pair may have multiple relationships [64]. Moreover, the integration of a Bayesian network with MIML was explored for relation extraction [65], further expanding and capabilities of the approach.

Although the bootstrapping method is intuitive and effective, it may introduce a large number of noisy instances during seed expansion, leading to semantic drift. To address it, a deep co-learning was proposed [66], and it was a semi-supervised end-to-end deep learning method for evaluating the credibility of Arabic Blogs. A coupled semi-supervised learning method was used to establish constraints between different categories of extraction templates [67]. This strategic implementation effectively curbed the generation of false templates, thereby bolstering the precision of relation extraction. Meanwhile, a method was proposed by combining matrix-vector recursive neural network (MV-RNN) with bootstrapping [68]. Here, through the tree structure in MV-RNN, the semantic information of the entire sentence could be extracted as relation classifier features, and it greatly improved the accuracy of the results avoiding the problem of requiring a large amount of corpus in MV-RNN. More recently, the integration of transfer learning, a popular machine learning strategy, into the semi-supervised learning, yielded a novel framework [69]. Applied within the context of low-resource entity and relation extraction in the scientific domain, this framework demonstrated satisfactory performance, underscoring its potential and versatility.

(3) Relation Extraction with Unsupervised Learning Method

Unsupervised learning-based relation extraction assumes that pairs of entities with the same semantic relation have similar contexts, and it transforms the relation extraction task into a clustering problem. Hence, it does not require manual corpus annotation, but the accuracy rate is relatively low.

Generally, it is implemented using various clustering algorithms. For example, the large pre-trained language model was used for adaptive clustering on contextualized relational features to improve computational performance in relation classification [70]. After taking the entity set in the Wikipedia entry as the object, and using the dependency features and shallow grammar templates, all semantic relationship instances corresponding to entities were extracted in a large-scale corpus by pattern clustering [71]. Meanwhile, the templates were extracted and aggregated from search engine summaries, and they were clustered to discover implicit semantic relationships represented by entity pairs [72]. To further elevate the efficacy of relational templates' clustering, a co-clustering algorithm was used, leveraging the dual nature of the dual of relational instances and relational templates. Moreover, the integration of a logistic regression model played a pivotal role in filtering representative extraction templates from the clustering results of relational templates [73].

For the above three basic types of machine learning, Table 4 offers a comprehensive comparison and analysis of the algorithms used in relation extraction. The table highlights the distinctions between classical algorithms, extraction ideas, human intervention levels, and extraction performance across different methods. Specifically, supervised and semi-supervised methods extract relationships by classifying learning sample data labels, while unsupervised-based methods cluster data and group related entities together to achieve relationship extraction. A noteworthy observation is that the performance of the extraction model improves with an increased corpus size and manual intervention. Taking into account the trade-off between performance and computational efforts, the semi-supervised learning methods emerge as favorable choices for relation extraction. It not only ensures better extraction performance but also avoids the limitation of requiring an extensive amount of manually annotated corpus. Moreover, Table 5 provides a comprehensive overview of the performance exhibited by various excellent relation extraction models on two famous datasets (NYT [74] and WebNLG [75]) in recent years. The F1-score achieved by these models indicate their effectiveness in relation extraction.

Table 4: Comparison of three kinds of machine learning-based relation extraction methods

Extraction method	Examples of learning algorithms	The idea of extraction	Manual intervention	Extraction performance
Supervised learning-based ones	SVM/ME/CRF /RNN/CNN	Classifying	More	High
Semi-supervised learning -based ones	Bootstrapping /Co-learning/MV-RNN	Classifying	Less	Medium
Unsupervised learning-based ones	Hierarchical Clustering /Co-clustering	Clustering	None	Low

Table 5: The top 10 relation extraction models from 2020 to 2022

Datasets	Model	F1-score (%)	Year	Platform
NYT	UniRel [76]	93.7	2022	Github
	REBEL [77]	93.4	2021	Github/HuggingFace
	DIRECT [78]	92.5	2021	Github

(Continued)

Table 5 (continued)

Datasets	Model	F1-score (%)	Year	Platform
	PFN [79]	92.4	2021	Github
	SPN [80]	92.5	2023	Github
	TDEER [81]	92.5	2021	Github
	RIFRE [82]	92.0	2021	Github
	TPLinker [83]	91.9	2020	Github
	PCNN+RL+HME [84]	90.0	2020	Github
	CasRel [85]	89.6	2020	Github
WebNLG	UniRel [76]	94.7	2022	Github
	PFN [79]	93.6	2021	Github
	SPN [80]	93.4	2023	Github
	TDEER [81]	93.1	2021	Github
	RIFRE [82]	92.6	2021	Github
	TPLinker [83]	91.9	2020	Github
	CasRel [85]	91.8	2020	Github
	RIN [86]	90.1	2020	Github
	CGT [87]	83.4	2021	–
	JointER [88]	83.1	2020	Github

Notes: For more detailed and up-to-date information about the models than what is presented in the article, readers can refer to the following links: https://paperswithcode.com/sota/relation-extraction-on-nyt; https://paperswithcode.com/sota/relation-extraction-on-webnlg.

Focusing on the NYT dataset, UniRel [76] achieved the highest F1-score of 93.7%, followed by Relation Extraction By End-to-end Language generation (REBEL) [77] with 93.4%, and Djacency lIst oRiented rElational faCT (DIRECT) [78] at 92.5%. These models have exhibited robust competence in extracting relations from the NYT dataset.

For the WebNLG dataset, UniRel [78] maintains its lead, achieving the highest F1-score of 94.7%, followed by Partition Filter Network (PFN) [79] with 93.6%, and Set Prediction Networks (SPN) [80] with 93.4%. These models have exhibited remarkable performance in relation extraction from the WebNLG dataset. Furthermore, other models such as Translating Decoding Schema for Joint Extraction of Entities and Relations (TDEER) [81] and Representation Iterative Fusion based on Heterogeneous Graph Neural Network for Joint Entity and Relation Extraction (RIFRE) [82] also achieved comparable performance on both datasets, further highlighting the consistent excellence of these methodologies.

These excellent models utilize various techniques, including RNNs, partition-based methods, and joint learning frameworks. The impressive F1-score attained by these models underscores their ability to effectively extract relations from text across different datasets. This analysis showcases the advancements in relation extraction models and their potential applications in numerous NLP tasks.

3.1.3 Entity Linking

Due to the diversity of information expression, entity ambiguity remains a frequent and formidable hurdle in natural language understanding. For example, “apple”, can signify either a

fruit or the renowned technology company. However, when contextual information such as "Steve Jobs" is provided, it becomes evident that in the given text, the entity "apple" refers to the company, as illustrated in Fig. 11. The presence of contextual cues helps disambiguate the intended meaning of entities, aiding in enhancing the precision and comprehension of text.

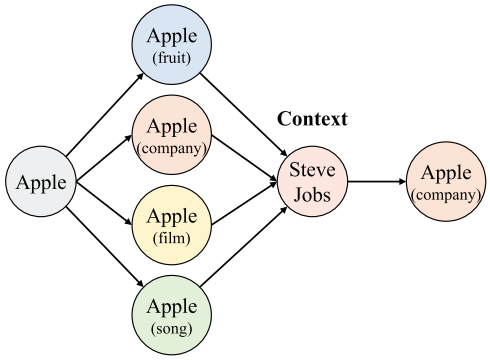

Figure 11: An example of entity linking

Entity linking is an effective disambiguation method. By linking entity mentions to the corresponding entities in the knowledge bases, this method conducts precise entity annotation within documents. This enables computers to attain a more profound grasp of the semantic information of the text, effectively addressing the problems of synonym and polysemy. Commonly, entity linking leverages well-established knowledge bases such as Wikipedia, DBPedia, and Freebase. Specifically, open knowledge bases like Baidu Encyclopedia and Interactive Encyclopedia also find prominent applications in Chinese entity linking. Generally speaking, entity linking includes two subtasks: entity recognition and entity disambiguation. By skillfully addressing both aspects, this technique contributes to a more comprehensive and nuanced understanding of textual content.

(1) Entity Recognition

Entity recognition aims to identify fragments of text, and it may link to specific entries in the knowledge bases, including a specific word or phrase. Typical entity types include place names, person names, institution names, times, dates, percentages, and amounts. With the development of Internet information, novel entity categories have surfaced in recent years, including movie titles and product names, reflecting the expanding landscape of entity recognition [89,90].

Currently, entity recognition technology is mainly based on statistical machine learning methods, treating the task as a sequence labeling problem. There are three main solutions: hidden Markov model (HMM), maximum entropy Markov model (MEMM), and CRF model. An overview of these models, their key evaluation criteria, and comparisons are delineated in Table 6. Over recent years, numerous studies have utilized these three models. For instance, a novel generative model was proposed [91], linking it to HMM while proving its generically identifiable nature without any observed training labels. However, HMM exhibits the tag bias problem by assuming that the current tag solely depends on the previous one. Additionally, its use of local normalization to compute the probability of observation series makes calculating the function complex, leading to reduced efficiency. MEMM addresses the tag bias problem but introduces the tag inconsistency issue. Furthermore, it often relies on manually designed features, which are pivotal for the model's performance and generalization ability, making feature selection a challenging task. CRF model is different from HMM and MEMM. By employing global modeling, CRF simultaneously considers the observation and labeling of the

entire sequence when calculating the probability. This enables the avoidance of label bias and local normalization problems found in HMM and MEMM. Moreover, the CRF model resolves the label inconsistency problem present in MEMM by modeling dependencies between observations and label sequences concurrently. Additionally, CRF utilizes complex and rich feature representations that capture dependencies between observed and labeled sequences, thereby enhancing model performance and generalization. Unlike MEMM, CRF model is less dependent on manually selected features and can learn feature weights that suit the task, thereby streamlining feature engineering. By overcoming some of the main shortcomings of HMM and MEMM models, CRF achieves superior performance in sequence labeling and has been widely used in the entity recognition field [92].

Table 6: Comparative analysis of three named entity recognition methods

Model	Decision condition	Feature selection	Whether to mark the deviation	Computational complexity
HMM	Mutually information probability	Limited	No	Low
MEMM	Conditional probability	Flexible	Yes	Low
CRF	Global probability	Flexible	No	High

(2) Entity Disambiguation

In the context of a specific entity mentioned in the text, entity disambiguation is mainly used to analyze the semantic information and select the corresponding entity from the candidate entities. This process depends on the candidate entity and the contextual information. Usually, given the ambiguity in natural language, there are numerous candidate entities vying for consideration. Methods of entity disambiguation are mainly based on supervised learning and unsupervised learning.

Most works adopt supervised learning methods for disambiguation and use training data to automatically design ranking models. The requisite training data for entity linking comprises an ordered list of all candidate entities associated with the target mentioned in a given context. In this list, the first entity is usually the one that the mention refers to in this context. By linearly combining features like entity popularity, semantic similarity, and connection between entities, a maximum margin-based data were employed to train feature weights, while achieving entity disambiguation through a ranking model [93]. Moreover, two machine learning sorting-based methods using listwise and pairwise were presented to implement entity disambiguation, outperforming traditional disambiguation methods [94]. The listwise method is from the LiNet algorithm, which uses the ordered list as a training instance to obtain a sorting model. The essence of the pairwise method is to transform the sorting problem into a classification problem. It combines the items in the ordered list into pairs and constructs training instances according to the relative positional relationship between the items to develop a sorting perception. Furthermore, recent developments have seen the combination of supervised learning with graph theory to address the entity disambiguation task [95], showcasing the innovative fusion of established techniques to enhance disambiguation accuracy.

On the other side, unsupervised learning algorithms have been used in entity disambiguation. For example, a clustering-based personal name disambiguation system was proposed to extract personal attributes and social relations between entities from text, subsequently mapping them onto

an undirected weighted graph [96]. Clustering algorithms were then used to cluster these graphs, each cluster contained all web pages that directed a person. Significant models have been shown in Table 7 in the field of entity disambiguation, where datasets are ACE2004 [97] and AIDA-CoNLL [98]. These models focus on disambiguating entities, which is crucial for accurately identifying the intended meaning of ambiguous terms.

For the ACE 2004 dataset, Entity Disambiguation by Reasoning over a Knowledge Base (KBED) [99] stands out with the highest F1-score of 93.4%, followed by LUKE[confidence-order] [100] with 91.9%, and Representation and Fine-grained typing for Entity Disambiguation (ReFinED) [101] at 91.6%. These models have shown strong performance in entity disambiguation on the ACE 2004 dataset.

Table 7: The top 5 entity disambiguation models

Datasets	Model	F1-score (%)	Year	Platform
ACE 2004	KBED [99]	93.4	2022	Github
	LUKE[confidence-order] [100]	91.9	2022	Github
	ReFinED [101]	91.6	2022	Github
	NER4EL [102]	91.3	2021	Github
	GENRE [103]	90.1	2020	–
AIDA-CoNLL	LUKE[confidence-order] [100]	95.0	2022	Github
	DCA-SL+Triples [104]	94.9	2020	Github
	DeepType [105]	94.9	2018	Github
	NTEE [106]	94.7	2017	Github
	DCA-SL [107]	94.6	2019	Github

Notes: For more detailed and up-to-date information about the models than what is presented in the article, readers can refer to the following links: https://paperswithcode.com/sota/entity-disambiguation-on-ace2004; https://paperswithcode.com/sota/entity-disambiguation-on-aida-conll.

Regarding the AIDA-CoNLL dataset, LUKE[confidence-order] [100] maintains its prominence with the highest F1-score of 95.0%, closely followed by Dynamic Context Augmentation (DCA)-Supervised Learning (SL)+Triples [104] and DeepType [105] with F1-score of 94.9%. These models have demonstrated excellent performance in disambiguating entities in the AIDA-CoNLL dataset. It is worth mentioning that LUKE[confidence-order] [100] is the only model present in both datasets, indicating its robustness and effectiveness across different evaluation scenarios.

The top-performing models employ diverse techniques such as knowledge-based methods, confidence ordering, and deep learning approaches. Their high F1-score emphasize their proficiency in accurately disambiguating entities and determining their intended meanings in different contexts.

3.2 Ontology Learning

An ontology is a formalized specification of the shared conceptual model, and it defines the pattern layer of KG. The composition of an ontology as O, entails the components (C, *root*, R). Here, C is the set of upper-level concepts, the *root* is the root identifier, and R is the binary relationship on C, including the synonymy relation and the hyponymy relation as Fig. 12 [108]. The purpose of an ontology lies in establishing an organized framework, thereby facilitating the organization and

categorization of concepts within the KG, enabling efficient retrieval and navigation of knowledge stored within. Generally speaking, ontology construction has three ways: manual construction, automatic construction, and semi-automatic construction. Manual construction method necessitates the participation of domain experts and entails the utilization of dedicated ontology editing tools. However, this approach tends to demand significant human and material resources, leading to scalability issues. Hence, it cannot keep up with the rapid development and update of Internet data. Ontology automatic construction or ontology learning includes extraction of concept, synonymy relation, and hyponymy relation from bottom to top. This process is predominantly automated, often relying on data-driven or cross-language knowledge linkage techniques that are grounded in machine learning principles. For instance, in order to address the ontology automation issues in the semantic web, ontology learning was achieved through the presentation of automatic or semi-automatic, aimed at either generating new ontology resources or repurposing existing ones [109].

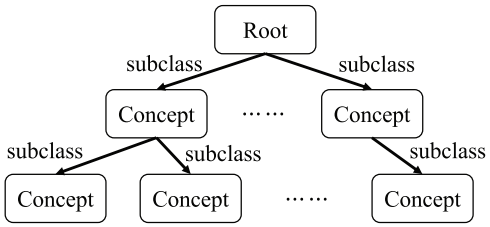

Figure 12: The flowchart of ontology modeling

Concept is the most basic unit of human comprehension. The common methods of concept extraction include linguistic methods [110], statistical methods [111], and machine learning methods. Within the domain of machine learning-based concept extraction, prevalent methods center around SVM [49], HMM [91], bootstrapping [112], and clustering. These techniques entail the extraction of pertinent categorical attributes from the dataset. For example, bootstrapping was used to automatically extract domain vocabulary from large-scale, unlabeled real corpus [112]. A bootstrapping-based seed expansion mechanism was developed to realize the automatic extraction of domain seed words [113]. Using the concept of clusters, the multi-topic extraction algorithm was introduced to acquire multiple topics by clustering concepts [114].

Relationship extraction between concepts mainly refers to the synonymy and the hyponymy relation. Synonymy relation extraction is to examine the degree of probability that any two entities belong to the same conceptual level. For example, entities like "Beijing" and "Shanghai", both denoting city names, exhibit a synonymy relation. Hyponymy relation extraction gauges the probability that any two entities establish a hierarchical relation, where one serves as a subtype of the other. For example, "Beijing" is the hyponymy of "city". For the supervised learning method, a novel approach was proposed for synonym identification using the principle of distributional similarity [115]. Compared to the traditional similarity models, the experimental results showed that a satisfactory performance was achieved while increasing by over 120% on the F_1 metric. A syntactic parser was used to construct a syntax tree, and the contextual features of concepts were used as concept attributes to generate concept lattices [116]. This led to the establishment of a partial order relationship of concept lattices, which subsequently formed the conceptual hierarchy of ontology. For the unsupervised learning method, a method was proposed to learn taxonomy from a collection of text documents, each dedicated to describing a distinct concept [117]. Specifically, with the continuous development of online encyclopedias, the machine learning method using the linked data has gradually become an efficient strategy for hyponymy relation extraction. Machine learning techniques were used to

explicitly represent the meaning of any text as a weighted vector of Wikipedia-based concepts. The cosine of the angle between these vectors was then calculated to measure the similarity between concepts or texts, effectively fostering the extraction of hyponymy relations [118].

3.3 Knowledge Reasoning

Building upon the foundation of existing entities and relationships, knowledge reasoning amis to mine implicit connections between entities through a sophisticated reasoning mechanism. The ultimate goal is to enhance and amplify the original KG by unveiling implicit relationships. For instance, as illustrated in Fig. 13, if the KG contains the information "Lion is-a animal" and "Animal can run", it becomes possible to infer the knowledge that "Lion can run" through logical reasoning. This process stands as a testament to the potential of knowledge reasoning, allowing the enrichment and augmentation of the KG through the generation of novel insights from preexisting information. This iterative process culminates in heightened completeness and a more profound grasp of semantic understanding.

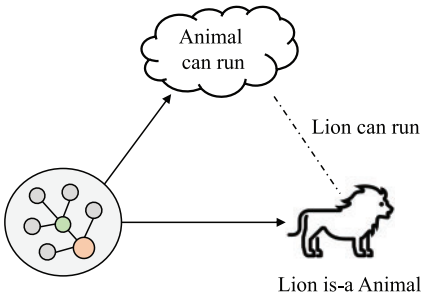

Figure 13: An example of knowledge reasoning

Traditional knowledge reasoning is mainly based on logical reasoning methods, including predicate logic, description logic, rule-based reasoning, and others. The predicate logic method is generally designed for simple entity relations. This method takes propositions as the fundamental units of reasoning, where atomic propositions are generally decomposed into two parts: individual words and predicates. Description logic can be used for complex relationships between entities. The typical path ranking algorithm plays a pivotal role in establishing rule-based reasoning. Distinctive relationship paths were used as one-dimensional features and the classification feature vector of the relationship was constructed by counting a large number of relationship paths in the graph [119]. However, it is important to acknowledge that logical reasoning necessitates the formulation of rules, a task that often proves computationally onerous and encounters challenges posed by data sparsity.

Link prediction is a new type of knowledge reasoning method under the statistical machine learning framework. It is to predict the possibility of the linked relationship between two unlinked nodes through the known nodes and link information in the KG, while discovering the implicit relationship between entities. The link prediction in the KG is generally realized by the representation learning method on the basis of the triples (e_i, r_k, e_j) used to constitute the knowledge. For the representation learning method, the semantic information of an entity is represented as dense low-dimensional real-valued vectors. Within this space, gauging the semantic similarity between objects is facilitated by mathematical methods such as cosine distance and Euclidean distance [120].

Recently, a large number of works on the representation learning-based link prediction have been proposed. One illustrative instance is the structured embedding (SE) method. This method projects

the entity vectors e_i and e_j through the two relation matrices of the inter-entity relation r_k to the corresponding space of r_k [121]. Then, the distance between two projection vectors on this space was calculated to gauge the confidence of r_k. The single layer model (SLM) was proposed through the application of a single-layer neural network [122]. This network employs nonlinear operation to define a scoring function for each triple. fostering the synergistic representation of the semantic connection between entities and relationships to reason about unknown relationships. While the accuracy of results exhibited significant enhancement over traditional methods, the adoption of nonlinear operations inevitably led to heightened computational complexity [122]. Another innovative contribution is the semantic matching energy (SME) model, which relies on low-dimensional vectors to represent entities and relationships [123]. Multiple projection matrices are employed to represent the connections between entities and relationships. The latent factor model (LFM), delves into a relationship-centered bilinear transformation. This transformation encapsulates the semantic connection between entities and relationships [124]. The RESACL model was presented as a typical knowledge representation method through matrix decomposition [125]. In this study, all triples (e_i, r_k, e_j) were represented as a large tensor, wherein the presence or absence of a triple dictates the value at the corresponding tensor position. Through the tensor decomposition algorithm, the tensor value corresponding to each triple in the tensor could be decomposed into entity and relation representations. Based on the characteristic of translation invariance of the word vector space, TransE model presents a pioneering perspective [126]. As depicted in Fig. 14, in this model, the semantic relation r_k between entities e_i and e_j was regarded as some kind of translation vector, and it actually was the translation from the head entity e_i to the tail entity e_j. Demonstrating substantial improvements in establishing intricate semantic connections within vast, sparse KGs, TransE has solidified its status as a pivotal model in this domain. Recent developments have exhibited a growing interest in the TransE model, exemplified by the increased attention it has garnered in studies [127,128].

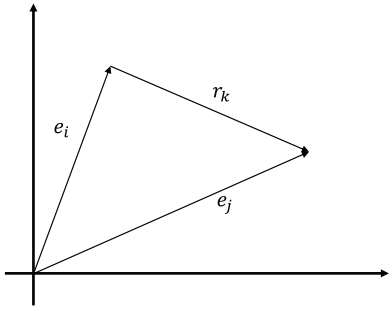

Figure 14: A brief description of the TransE model

4 The Future Challenges of Constructing a Large-Scale Knowledge Graph

Amidst the era of big data, the KG provides a new learning paradigm to efficiently organize, manage and understand massive amounts of information, while presenting this data in a manner closely aligned with human cognition. Hence, it promotes rapid advancements across many fields such as information retrieval, knowledge recommendation, and others. Although significant improvements have been achieved in the research of KG, the recently developed machine learning technologies and methods in big data analysis still cannot effectively match the demands of exploiting and using KG due to the complexity of real world application scenarios. In effect, a myriad of technical challenges persist, necessitating adept resolution to release the full potential of large-scale KG as a powerful methodology

for intelligent information service. Here, we summarize the future research trends and challenges of machine learning-driven large-scale KG construction from three aspects.

4.1 Relation Extraction

On one hand, the current focus of relation extraction predominantly revolves around the monolingual text. However, in the real-world, factual knowledge finds its repository in diverse sources, such as multilingual texts, pictures, audio, and video. Expanding the horizons of relation extraction to encompass these various sources stands as a promising avenue for future exploration, heralding the potential to broaden the spectrum of extracted relations and extend the scope of knowledge coverage [56]. On the other hand, for the deep learning-based relation extraction methods, the integration of syntactic trees via neural network models yields the effective amalgamation of syntactic information. However, it also leads to the introduction of a large amount of noise, which poses an impact on the accuracy of the model [57,60]. Constructing multiple possible syntactic trees of sentences and fusing them for relation extraction may be a development prospect. Furthermore, the open field-based relationship extraction is constantly updated and iterative. How to introduce deep learning models to achieve rapid learning of new relationships and knowledge is also a problem that needs to be explored [129].

4.2 Link Prediction

Link prediction plays an important role in the design of KG, serving as a critical component in inferring absent relationships. For example, as shown in Fig. 15, where known relationships between entities A and B, B and C, and C and D, the link prediction method enables the inference of the relationship type between entities A and D. By leveraging the existing relationships within the KG, this technique enables the identification and prediction of unobserved connections, thereby enhancing the overall comprehension and knowledge extraction from the graph. Generally, the implementation of link prediction is mainly based on triples used to constitute knowledge. However, the types of knowledge are rich and diverse, and some complex knowledge cannot be directly represented by triples. Hence, different knowledge representation methods need to be set up for different scenarios. For instance, considering the temporal dynamism of factual values, a compound value type structure has been introduced, involving auxiliary nodes to represent n-order relations and temporal attributes for facts [130]. Multiple n-order relations can be represented by a single $(n + 1)$-order tensor, which is solved by higher-order tensor decomposition using the RESACL model [131]. The representation of learning-based link prediction is still in the initial stage of exploration [132]. It achieves unsatisfactory performance on large-scale KG with strong sparsity and the representation of low-frequency entities and relationships. It is urgent to design a more efficient online learning scheme for KG. Concurrently, the domain of network embedding-based algorithms, including those grounded in graph neural networks (GNN), has showcased compelling computational prowess in task completion. Specifically, the attention mechanism-based heterogeneous GNN is conducive to capturing information of various semantics in KGs [133]. Additionally, the introduction of the multi-scale dynamic convolutional network (M-DCN) has provided a framework for representing KG embeddings [134]. Therefore, how to creatively investigate those algorithms in the achievement of link prediction for KG is also an interesting direction [135,136].

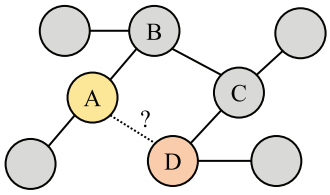

Figure 15: Node roles for link prediction

4.3 Construction of Industrial Knowledge Graph

In a broader context, the construction of knowledge graphs (KGs) has spanned general domains, and the corresponding theoretical investigations have followed suit. However, the construction of KG for a specific field, especially in industry, has attracted less attention in recent years. General KGs emphasize the entity layer and it is difficult to generate a global ontology pattern. Notably, there are many differences between industrial KG and general KG [137]. The industrial KG has a clear industry background, while the entities have rich data patterns. Furthermore, the industrial KGs need comprehensive consideration of personnel at various hierarchical levels. This has led to new diverse challenges for researchers focusing on the design of a large-scale industrial KG. Thus, it needs to be explored in this direction, while presenting some new machine learning-driven methods in these fields [138,139].

Over the last year, significant advancements have been made in NLP tasks with the emergence of large language models (LLMs) like ChatGPT [140], Dolly [141], and LLaMA [142]. Despite their achievements, LLMs are black-box models, lacking transparency in tracking their search process. On the other hand, KG offers a high level of professional feasibility, providing more accurate answers and better interpretability compared to LLMs. Leveraging the respective advantages of LLMs and KG, we can utilize LLMs for data annotation and data enhancement in the future, facilitating the rapid application of industrial KGs [143]. Meanwhile, industrial KG can serve as an external knowledge base, enabling the introduction of specified constraints to LLMs [144]. This allows for controlled content generation and enhances the adaptability of LLMs within specific industrial fields.

5 Conclusion

The centrality of KGs within the realm of next-generation search engines has established them as a pivotal focal point within the sphere of intelligent information processing, coinciding with the advent of the big data era. With the ever-increasing demand for processing performance in consideration of complex application scenarios, there are many exploratory works to be explored in the KG construction. In this article, we comprehensively review the key technologies in KG construction from the perspective of machine learning-related implementation methods. We are concerned with the core construction algorithms of KG in three aspects, including entity learning, ontology learning, and knowledge reasoning. Especially, the machine learning-driven algorithms for entity extraction, relation extraction, entity linking, and link prediction are deeply analyzed. In addition, considering the current development level of machine learning methods, we summarize some key problems and possible research trends in the construction of large-scale KG to serve as an impetus for researchers to work in the future.

Acknowledgement: None.

Funding Statement: This work was supported in part by the Beijing Natural Science Foundation under Grants L211020 and M21032, in part by the National Natural Science Foundation of China under Grants U1836106 and 62271045, and in part by the Scientific and Technological Innovation Foundation of Foshan under Grants BK21BF001 and BK20BF010.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Z. Zhao, X. Luo; data collection: Z. Zhao, M. Chen, L. Ma; analysis and interpretation of results: Z. Zhao, X. Luo, M. Chen; draft manuscript preparation: Z. Zhao, X. Luo, L. Ma. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The presented data are open-sourced and can be accessed through the "Web of Science" and "Papers with Code website".

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Ragavan, N., Rubavathi, C. Y. (2022). A novel big data storage reduction model for drill down search. *Computer Systems Science and Engineering, 41(1),* 373–387. https://doi.org/10.32604/csse.2022.020452
2. Singha, A. (2012). Official Google blog: Introducing the knowledge graph: Things not strings. http://googleblog.blogspot.pt/2012/05/introducing-knowledge-graph-things-not.html
3. Kshetri, N. (2022). Web 3.0 and the metaverse shaping organizations' brand and product strategies. *IT Professional, 24(2),* 11–15.
4. Greenbaum, D. (2022). The virtual worlds of the metaverse. *Science, 377(6604),* 377.
5. Zamini, M., Reza, H., Rabiei, M. (2022). A review of knowledge graph completion. *Information, 13(8),* 396.
6. Miller, G. A. (1995). WordNet: A lexical database for English. *Communications of the ACM, 38(11),* 39–41.
7. Nickel, M., Murphy, K., Tresp, V., Gabrilovich, E. (2015). A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE, 104(1),* 11–33.
8. Sowa, J. F. (2014). *Principles of semantic networks: Explorations in the representation of knowledge*. San Mateo, CA, USA: Morgan Kaufmann.
9. Hoffart, J., Suchanek, F. M., Berberich, K., Lewis-Kelham, E., De Melo, G. et al. (2011). YAGO2: Exploring and querying world knowledge in time, space, context, and many languages. *Proceedings of the 20th International Conference Companion on World Wide Web*, pp. 229–232. New York, NY, USA, Association for Computing Machinery.
10. Opdahl, A. L., Al-Moslmi, T., Dang-Nguyen, D. T., Gallofré Ocaña, M., Tessem, B. et al. (2022). Semantic knowledge graphs for the news: A review. *ACM Computing Surveys, 55(7),* 1–38.
11. McGuinness, D. L., van Harmelen, F., Deborah, L., Frank (2004). OWL web ontology language overview. *W3C Recommendation, 10(10),* 1–12.
12. Lygerakis, F., Kampelis, N., Kolokotsa, D. (2022). Knowledge graphs' ontologies and applications for energy efficiency in buildings: A review. *Energies, 15(20),* 7520.
13. Yu, H., Li, H., Mao, D., Cai, Q. (2021). A domain knowledge graph construction method based on Wikipedia. *Journal of Information Science, 47(6),* 783–793.

14. Srirangam, V. K., Reddy, A. A., Singh, V., Shrivastava, M. (2019). Corpus creation and analysis for named entity recognition in Telugu-English code-mixed social media data. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pp. 183–189. Florence, Italy, Association for Computational Linguistics.
15. Sykes, D., Grivas, A., Grover, C., Tobin, R., Sudlow, C. et al. (2021). Comparison of rule-based and neural network models for negation detection in radiology reports. *Natural Language Engineering, 27(2),* 203–224.
16. Whitelaw, C., Kehlenbeck, A., Petrovic, N., Ungar, L. (2008). Web-scale named entity recognition. *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, pp. 123–132. Singapore, Association for Computing Machinery.
17. Jain, A., Pennacchiotti, M. (2010). Open entity extraction from web search query logs. *Proceedings of the 23rd International Conference on Computational Linguistics*, pp. 510–518. Beijing, China.
18. Liu, X., Fan, J., Ma, H., Yang, Z. (2022). Research on application of intelligent corpus annotation of entity extraction with construction of knowledge graph. *Mathematical Problems in Engineering, 2022,* 2552331.
19. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K. et al. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research, 12,* 2493–2537.
20. Zhao, Z., Yang, Z., Luo, L., Zhang, Y., Wang, L. et al. (2016). ML-CNN: A novel deep learning based disease named entity recognition architecture. *Proceedings of the 2016 IEEE International Conference on Bioinformatics and Biomedicine*, pp. 794. Shenzhen, China, IEEE.
21. Gui, T., Ma, R., Zhang, Q., Zhao, L., Jiang, Y. G. et al. (2019). CNN-based Chinese NER with lexicon rethinking. *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 4982–4988. Macao, China, IEEE.
22. Cho, M., Ha, J., Park, C., Park, S. (2020). Combinatorial feature embedding based on CNN and LSTM for biomedical named entity recognition. *Journal of Biomedical Informatics, 103,* 103381.
23. Tjong Kim Sang, E. F., de Meulder, F. (2003). Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL*, pp. 142–147. Edmonton, AB, Canada, Computational Linguistics in Flanders.
24. Walker, C., Strassel, S., Medero, J., Maeda, K. (2006). *ACE 2005 multilingual training corpus*. Philadelphia, USA: Linguistic Data Consortium, University of Pennsylvania.
25. Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., Weischedel, R. (2006). OntoNotes: The 90% solution. *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pp. 57–60. Boulder, CO, USA, Association for Computational Linguistics.
26. Wang, X., Jiang, Y., Bach, N., Wang, T., Huang, Z. et al. (2021). Automated concatenation of embeddings for structured prediction. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pp. 2643–2660. Bangkok, Thailand, Association for Computational Linguistics.
27. Zhou, W., Chen, M. (2021). Learning from noisy labels for entity-centric information extraction. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 5381–5392. Punta Cana, Dominican Republic, Association for Computational Linguistics.
28. Liu, T., Jiang, Y. E., Monath, N., Cotterell, R., Sachan, M. (2022). Autoregressive structured prediction with language models. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 993–1005. Abu Dhabi, United Arab Emirates, Association for Computational Linguistics.
29. Schweter, S., Akbik, A. (2020). FLERT: Document-level features for named entity recognition. arXiv preprint arXiv:2011.06993.
30. Ye, D., Lin, Y., Li, P., Sun, M. (2022). Packed levitated marker for entity and relation extraction. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pp. 4904–4917. Dublin, Ireland, Association for Computational Linguistics.

31. Yamada, I., Asai, A., Shindo, H., Takeda, H., Matsumoto, Y. (2020). LUKE: Deep contextualized entity representations with entity-aware self-attention. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 6442–6454. Association for Computational Linguistics.
32. Wang, X., Jiang, Y., Bach, N., Wang, T., Huang, Z. et al. (2021). Improving named entity recognition by external context retrieving and cooperative learning. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pp. 1800–1812. Bangkok, Thailand, Association for Computational Linguistics.
33. Hanh, T. T. H., Doucet, A., Sidere, N., Moreno, J. G., Pollak, S. (2021). Named entity recognition architecture combining contextual and global features. *Proceedings of the 23rd International Conference on Asia-Pacific Digital Libraries*, pp. 264–276. Springer.
34. Shahzad, M., Amin, A., Esteves, D., Ngonga Ngomo, A. C. (2021). InferNER: An attentive model leveraging the sentence-level information for named entity recognition in Microblogs. *Proceedings of the 34th International Florida Artificial Intelligence Research Society Conference*, Miami, FL, USA, Library Press.
35. Zhong, Z., Chen, D. (2021). A frustratingly easy approach for entity and relation extraction. *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 50–61. Punta Cana, Dominican Republic, Association for Computational Linguistics.
36. Shen, Y., Tan, Z., Wu, S., Zhang, W., Zhang, R. et al. (2023). PromptNER: Prompt locating and typing for named entity recognition. *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics Volume 1: Long Papers*, pp. 12492–12507. Toronto, Canada: Association for Computational Linguistics.
37. Shen, Y., Wang, X., Tan, Z., Xu, G., Xie, P. et al. (2022). Parallel instance query network for named entity recognition. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pp. 947–961. Dublin, Ireland, Association for Computational Linguistics.
38. Shen, Y., Song, K., Tan, X., Li, D., Lu, W. et al. (2023). DiffusionNER: Boundary diffusion for named entity recognition. *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics Volume 1: Long Papers*, pp. 3875–3890. Toronto, Canada: Association for Computational Linguistics.
39. Li, X., Feng, J., Meng, Y., Han, Q., Wu, F. et al. (2020). A unified MRC framework for named entity recognition. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5849–5859. Association for Computational Linguistics.
40. Shen, Y., Ma, X., Tan, Z., Zhang, S., Wang, W. et al. (2021). Locate and label: A two-stage identifier for nested named entity recognition. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pp. 2782–2794. Association for Computational Linguistics.
41. Jiang, H., Wang, G., Chen, W., Zhang, C., Karlsson, B. F. (2021). BoningKnife: Joint entity mention detection and typing for nested NER via prior boundary knowledge. arXiv preprint arXiv:2107.09429.
42. Yu, J., Bohnet, B., Poesio, M. (2020). Named entity recognition as dependency parsing. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 6470–6476. Association for Computational Linguistics.
43. Shibuya, T., Hovy, E. (2020). Nested named entity recognition via second-best sequence learning and decoding. *Transactions of the Association for Computational Linguistics, 8,* 605–620.
44. Li, X., Sun, X., Meng, Y., Liang, J., Wu, F. et al. (2020). Dice loss for data-imbalanced NLP tasks. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 465–476. Association for Computational Linguistics.
45. Zhu, E., Li, J. (2022). Boundary smoothing for named entity recognition. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 7096–7108. Dublin, Ireland, Association for Computational Linguistics.

46. Hu, J., Shen, Y., Liu, Y., Wan, X., Chang, T. H. (2022). Hero-Gang neural model for named entity recognition. *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1924–1936. Seattle, WA, USA, Association for Computational Linguistics.
47. Xu, L., Jie, Z., Lu, W., Bing, L. (2021). Better feature integration for named entity recognition. *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3457–3469. Punta Cana, Dominican Republic, Association for Computational Linguistics.
48. Li, J., Fei, H., Liu, J., Wu, S., Zhang, M. et al. (2022). Unified named entity recognition as word-word relation classification. *Proceedings of the 36th AAAI Conference on Artificial Intelligence*, pp. 10965–10973. Association for the Advance of Artificial Intelligence.
49. Plaza-del Arco, F. M., Molina-González, M. D., Martin, M., Ureña-López, L. A. (2019). SINAI at SemEval-2019 task 6: Incorporating lexicon knowledge into SVM learning to identify and categorize offensive language in social media. *Proceedings of the 13th International Workshop on Semantic Evaluation*, pp. 735–738. Minneapolis, MN, USA, Association for Computational Linguistics.
50. Lv, C., Pan, D., Li, Y., Li, J., Wang, Z. (2021). A novel Chinese entity relationship extraction method based on the bidirectional maximum entropy Markov model. *Complexity, 2021,* 1–8.
51. Li, J., Luo, X., Ma, H., Zhao, W. (2023). A hybrid deep transfer learning model with kernel metric for COVID-19 pneumonia classification using chest CT images. *IEEE/ACM Transactions on Computational Biology and Bioinformatics, 20(4),* 2506–2517.
52. Zhang, H., Hou, S., Xia, X. (2012). A novel convolution kernel model for Chinese relation extraction based on semantic feature and instances partition. *Proceedings of the 5th International Symposium on Computational Intelligence and Design*, pp. 411–414. Hangzhou, China, IEEE.
53. Sun, L., Han, X. (2014). A feature-enriched tree kernel for relation extraction. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pp. 61–67. Baltimore, Maryland, Association for Computational Linguistics.
54. Sobhana, N., Ghosh, S., Mitra, P. (2012). Entity relation extraction from geological text using conditional random fields and subsequence kernels. *Proceedings of the Annual IEEE India Conference*, pp. 832–840. Kochi, Kerala, India, IEEE.
55. Luo, X., Sun, J., Wang, L., Wang, W., Zhao, W. et al. (2018). Short-term wind speed forecasting via stacked extreme learning machine with generalized correntropy. *IEEE Transactions on Industrial Informatics, 14(11),* 4963–4971.
56. Kambar, M. E. Z. N., Esmaeilzadeh, A., Heidari, M. (2022). A survey on deep learning techniques for joint named entities and relation extraction. *Proceedings of the IEEE World AI IoT Congress*, pp. 218–224. Seattle, WA, USA, IEEE.
57. Socher, R., Huval, B., Manning, C. D., Ng, A. Y. (2012). Semantic compositionality through recursive matrix-vector spaces. *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 1201–1211. Jeju Island, Korea, Association for Computational Linguistics.
58. Kim, Y. (2014). Convolutional neural networks for sentence classification. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 1746–1751. Doha, Qatar, Association for Computational Linguistics.
59. Li, Y., Long, G., Shen, T., Zhou, T., Yao, L. et al. (2020). Self-attention enhanced selective gate with entity-aware embedding for distantly supervised relation extraction. *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, pp. 8269–8276. New York, USA, Association for the Advance of Artificial Intelligence.
60. dos Santos, C., Xiang, B., Zhou, B. (2015). Classifying relations by ranking with convolutional neural networks. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and*

the 7th International Joint Conference on Natural Language Processing, pp. 626–634. Beijing, China, Association for Computational Linguistics.

61. Yin, Z., Wu, S., Yin, Y., Luo, W., Luo, Z. et al. (2019). Relation classification in scientific papers based on convolutional neural network. *Proceedings of the 8th CCF International Conference on Natural Language Processing and Chinese Computing*, pp. 242–253. Dunhuang, China, Springer.
62. Zhang, H., Feng, Y., Hao, W., Chen, G., Jin, D. (2017). Relation extraction with deep reinforcement learning. *IEICE Transactions on Information and Systems, 100(8),* 1893–1902.
63. Takamatsu, S., Sato, I., Nakagawa, H. (2012). Reducing wrong labels in distant supervision for relation extraction. *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pp. 721–729. Jeju Island, Korea, Association for Computational Linguistics.
64. Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L., Weld, D. S. (2011). Knowledge-based weak supervision for information extraction of overlapping relations. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 541–550. Portland, OR, USA, Association for Computational Linguistics.
65. Surdeanu, M., Tibshirani, J., Nallapati, R., Manning, C. D. (2012). Multi-instance multi-label learning for relation extraction. *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 455–465. Jeju Island, Korea, Association for Computational Linguistics.
66. Helwe, C., Elbassuoni, S., Al Zaatari, A., El-Hajj, W. (2019). Assessing Arabic weblog credibility via deep co-learning. *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pp. 130–136. Florence, Italy, Association for Computational Linguistics.
67. Carlson, A., Betteridge, J., Wang, R. C., Hruschka, E. R. Jr, Mitchell, T. M. (2010). Coupled semi-supervised learning for information extraction. *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining*, pp. 101–110. New York, NY, USA, Association for Computing Machinery.
68. Jianshu, J., Guang, C., Chunyun, Z. (2014). A bootstrapping and MV-RNN mixed method for relation extraction. *Proceedings of the 4th IEEE International Conference on Network Infrastructure and Digital Content*, pp. 117–120. Beijing, China, IEEE.
69. Wang, H., Mao, X. L., Heyan, H. (2022). A semi-supervised transfer learning framework for low resource entity and relation extraction in scientific domain. *Proceedings of the 3rd Workshop on Extraction and Evaluation of Knowledge Entities from Scientific Documents*, pp. 41–47. Cologne, Germany, CEUR-WS.
70. Hu, X., Wen, L., Xu, Y., Zhang, C., Yu, S. et al. (2020). SelfORE: Self-supervised relational feature learning for open relation extraction. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 3673–3682. Association for Computational Linguistics.
71. Yan, Y., Okazaki, N., Matsuo, Y., Yang, Z., Ishizuka, M. (2009). Unsupervised relation extraction by mining Wikipedia texts using information from the web. *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pp. 1021–1029. Singapore, Association for Computational Linguistics.
72. Bollegala, D., Matsuo, Y., Ishizuka, M. (2009). Measuring the similarity between implicit semantic relations using web search engines. *Proceedings of the 2nd ACM International Conference on Web Search and Data Mining*, pp. 104–113. Barcelona, Spain, Association for Computing Machinery.
73. Bollegala, D. T., Matsuo, Y., Ishizuka, M. (2010). Relational duality: Unsupervised extraction of semantic relations between entities on the web. *Proceedings of the 19th International Conference of World Wide Web*, pp. 151–160. Raleigh, NC, USA, Association for Computing Machinery.
74. Riedel, S., Yao, L., McCallum, A. (2010). Modeling relations and their mentions without labeled text. *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 148–163. Barcelona, Spain, Springer.

75. Gardent, C., Shimorina, A., Narayan, S., Perez-Beltrachini, L. (2017). Creating training corpora for NLG micro-planners. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 179–188. Vancouver, Canada, Association for Computational Linguistics.
76. Tang, W., Xu, B., Zhao, Y., Mao, Z., Liu, Y. et al. (2022). UniRel: Unified representation and interaction for joint relational triple extraction. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 7087–7099. Abu Dhabi, United Arab Emirates, Association for Computational Linguistics.
77. Huguet Cabot, P. L., Navigli, R. (2021). REBEL: Relation extraction by end-to-end language generation. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 2370–2381. Punta Cana, Dominican Republic, Association for Computational Linguistics.
78. Zhao, F., Jiang, Z., Kang, Y., Sun, C., Liu, X. (2021). Adjacency list oriented relational fact extraction via adaptive multi-task learning. *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 3075–3087. Association for Computational Linguistics.
79. Yan, Z., Zhang, C., Fu, J., Zhang, Q., Wei, Z. (2021). A partition filter network for joint entity and relation extraction. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 185–197. Punta Cana, Dominican Republic, Association for Computational Linguistics.
80. Sui, D., Zeng, X., Chen, Y., Liu, K., Zhao, J. (2023). Joint entity and relation extraction with set prediction networks. *IEEE Transactions on Neural Networks and Learning Systems,* pp. 1–12. https://doi.org/10.1109/TNNLS.2023.3264735
81. Li, X., Luo, X., Dong, C., Yang, D., Luan, B. et al. (2021). TDEER: An efficient translating decoding schema for joint extraction of entities and relations, *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 8055–8064. Punta Cana, Dominican Republic, Association for Computational Linguistics.
82. Zhao, K., Xu, H., Cheng, Y., Li, X., Gao, K. (2021). Representation iterative fusion based on heterogeneous graph neural network for joint entity and relation extraction. *Knowledge-Based Systems, 219,* 106888.
83. Wang, Y., Yu, B., Zhang, Y., Liu, T., Zhu, H. et al. (2020). TPLinker: Single-stage joint extraction of entities and relations through token pair linking. *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 1572–1582. Barcelona, Spain, International Committee on Computational Linguistics.
84. Wang, J. (2020). RH-Net: Improving neural relation extraction via reinforcement learning and hierarchical relational searching. arXiv preprint arXiv:2010.14255.
85. Wei, Z., Su, J., Wang, Y., Tian, Y., Chang, Y. (2020). A novel cascade binary tagging framework for relational triple extraction. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1476–1488. Association for Computational Linguistics.
86. Sun, K., Mensah, R. Z. S., Mao, Y., Liu, X. (2020). Recurrent interaction network for jointly extracting entities and classifying relations. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 3722–3732. Association for Computational Linguistics.
87. Ye, H., Zhang, N., Deng, S., Chen, M., Tan, C. et al. (2021). Contrastive triple extraction with generative transformer. *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence*, pp. 14257–14265. Association for the Advance of Artificial Intelligence.
88. Yu, B., Zhang, Z., Shu, X., Liu, T., Wang, Y. et al. (2020). Joint extraction of entities and relations based on a novel decomposition strategy. *Proceedings of the 24th European Conference on Artificial Intelligence*, pp. 2282–2289. Santiago de Compostela, Spain, IOS Press.
89. Li, J., Sun, A., Han, J., Li, C. (2020). A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering, 34(1),* 50–70.
90. Cheng, J., Liu, J., Xu, X., Xia, D., Liu, L. et al. (2021). A review of Chinese named entity recognition. *KSII Transactions on Internet & Information Systems, 15(6),* 2012–2030.

91. Safranchik, E., Luo, S., Bach, S. (2020). Weakly supervised sequence tagging from noisy rules. *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, pp. 5570–5578. New York, NY, USA, Association for the Advance of Artificial Intelligence.
92. Khan, W., Daud, A., Shahzad, K., Amjad, T., Banjar, A. et al. (2022). Named entity recognition using conditional random fields. *Applied Sciences, 12(13),* 6391.
93. Shen, W., Wang, J., Luo, P., Wang, M. (2012). Linden: Linking named entities with knowledge base via semantic knowledge. *Proceedings of the 21st International Conference on World Wide Web*, pp. 449–458. Lyon, France, Association for Computing Machinery.
94. Zheng, Z., Li, F., Huang, M., Zhu, X. (2010). Learning to link entities with knowledge base. *Proceedings of the Annual Conference of the North American Chapter of the Association of Computational Linguistics*, pp. 483–491. Los Angeles, CA, USA, Association for Computational Linguistics.
95. Mihaljević, H., Santamaría, L. (2021). Disambiguation of author entities in ADS using supervised learning and graph theory methods. *Scientometrics, 126(5),* 3893–3917.
96. Emami, H. (2019). A graph-based approach to person name disambiguation in web. *ACM Transactions on Management Information Systems, 10(2),* 1–25.
97. Mitchell, A., Strassel, S., Huang, S., Zakhary, R. (2005). *ACE 2004 multilingual training corpus*. Philadelphia, USA: Linguistic Data Consortium, University of Pennsylvania.
98. Hoffart, J., Yosef, M. A., Bordino, I., Fürstenau, H., Pinkal, M. et al. (2011). Robust disambiguation of named entities in text. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 782–792. Edinburgh, Scotland, UK, Association for Computational Linguistics.
99. Ayoola, T., Fisher, J., Pierleoni, A. (2022). Improving entity disambiguation by reasoning over a knowledge base. *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2899–2912. Seattle, WA, USA, Association for Computational Linguistics.
100. Yamada, I., Washio, K., Shindo, H., Matsumoto, Y. (2022). Global entity disambiguation with BERT. *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3264–3271. Seattle, WA, USA, Association for Computational Linguistics.
101. Ayoola, T., Tyagi, S., Fisher, J., Christodoulopoulos, C., Pierleoni, A. (2022). ReFinED: An efficient zero-shot-capable approach to end-to-end entity linking. *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track*, pp. 209–220. Seattle, WA, USA, Association for Computational Linguistics.
102. Tedeschi, S., Conia, S., Cecconi, F., Navigli, R. (2021). Named entity recognition for entity linking: What works and what's next. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 2584–2596. Punta Cana, Dominican Republic, Association for Computational Linguistics.
103. de Cao, N., Izacard, G., Riedel, S., Petroni, F. (2021). Autoregressive entity retrieval. *Proceedings of the 9th International Conference on Learning Representations*, pp. 1–20.
104. Mulang', I. O., Singh, K., Prabhu, C., Nadgeri, A., Hoffart, J. et al. (2020). Evaluating the impact of knowledge graph context on entity disambiguation models. *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pp. 2157–2160. Ireland, Association for Computing Machinery.
105. Raiman, J., Raiman, O. (2018). DeepType: Multilingual entity linking by neural type system evolution. *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, pp. 5406–5413. New Orleans, LA, USA, Association for the Advance of Artificial Intelligence.
106. Yamada, I., Shindo, H., Takeda, H., Takefuji, Y. (2017). Learning distributed representations of texts and entities from knowledge base. *Transactions of the Association for Computational Linguistics, 5,* 397–411.
107. Yang, X., Gu, X., Lin, S., Tang, S., Zhuang, Y. et al. (2019). Learning dynamic context augmentation for global entity linking. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*

and the 9th International Joint Conference on Natural Language Processing, pp. 271–281. Hong Kong, China, Association for Computational Linguistics.

108. Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition, 5(2),* 199–220.
109. Liu, Y. (2019). *Enhancing ontology learning with machine learning and natural language processing techniques. (Ph.D. Thesis)*. Rensselaer Polytechnic Institute Troy, New York, USA.
110. Shamsfard, M., Barforoush, A. A. (2004). Learning ontologies from natural language texts. *International Journal of Human-Computer Studies, 60(1),* 17–63.
111. Cimiano, P., Staab, S. (2005). Learning concept hierarchies from text with a guided hierarchical clustering algorithm. *Proceedings of the ICML Workshop on Learning and Extending Lexical Ontologies with Machine Learning Methods*, pp. 1–10. New York, NY, USA, CiteSeer.
112. Chen, W., Zhu, J., Yao, T., Zhang, Y. (2003). Automatic learning field words by bootstrapping. *Proceedings of China National Conference on Computational Linguistics*, pp. 67–72. Beijing, China, Tsinghua University Press.
113. Ji, D., Zhao, S., Xiao, G. (2009). Chinese document re-ranking based on automatically acquired term resource. *Language Resources and Evaluation, 43(4),* 385–406.
114. Ma, J., Yongjun, Z., Zhijian, W. (2015). Multi-topic extraction algorithm based on concept clusters. *CAAI Transactions on Intelligent Systems, 10(2),* 261–266.
115. Hagiwara, M. (2008). A supervised learning approach to automatic synonym identification based on distributional features. *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Student Research Workshop*, pp. 1–6. Columbus, Ohio, USA, Association for Computational Linguistics.
116. Cimiano, P., Hotho, A., Stumme, G., Tane, J. (2004). Conceptual knowledge processing with formal concept analysis and ontologies, *Proceedings of the 2nd International Conference on Formal Concept Analysi*, pp. 189–207. Sydney, Australia, Springer.
117. Paukkeri, M. S., García-Plaza, A. P., Fresno, V., Unanue, R. M., Honkela, T. (2012). Learning a taxonomy from a set of text documents. *Applied Soft Computing, 12(3),* 1138–1148.
118. Gabrilovich, E., Markovitch, S. (2007). Computing semantic relatedness using Wikipedia-based explicit semantic analysis. *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, pp. 1606–1611. Hyderabad, India, Association for Computing Machinery.
119. Lao, N., Cohen, W. W. (2010). Relational retrieval using a combination of path-constrained random walks. *Machine Learning, 81(1),* 53–67.
120. Liu, Z., Sun, M., Lin, Y., Xie, R. (2016). Knowledge representation learning: A review. *Journal of Computer Research and Development, 53(2),* 247–261.
121. Bordes, A., Weston, J., Collobert, R., Bengio, Y. (2011). Learning structured embeddings of knowledge bases, *Proceedings of the 25th AAAI Conference on Artificial Intelligence*, pp. 301–306. San Francisco, CA, USA, Association for the Advance of Artificial Intelligence.
122. Socher, R., Chen, D., Manning, C. D., Ng, A. (2013). Reasoning with neural tensor networks for knowledge base completion. *Proceedings of the 27th Annual Conference on Neural Information Processing Systems*, pp. 926–934. Lake Tahoe, NV, USA, Curran Associates, Inc.
123. Bordes, A., Glorot, X., Weston, J., Bengio, Y. (2014). A semantic matching energy function for learning with multi-relational data: Application to word-sense disambiguation. *Machine Learning, 94(2),* 233–259.
124. Jenatton, R., Roux, N., Bordes, A., Obozinski, G. R. (2012). A latent factor model for highly multi-relational data. *Proceedings of the 26th Annual Conference on Neural Information Processing Systems*, pp. 3167–3175. Lake Tahoe, NV, USA, Curran Associates, Inc.

125. Nickel, M., Tresp, V., Kriegel, H. P., et al. (2011). A three-way model for collective learning on multi-relational data. *Proceedings of the 28th International Conference on Machine Learning*, pp. 809–816. Bellevue, WA, USA, Association for Computing Machinery.
126. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. *Proceedings of the 27th Annual Conference on Neural Information Processing Systems*, pp. 2787–2795. Lake Tahoe, NV, USA, Curran Associates, Inc.
127. Karetnikov, A., Ehrlinger, L., Geist, V. (2022). Enhancing TransE to predict process behavior in temporal knowledge graphs. *Proceedings of the 33rd International Conference on Database and Expert Systems Applications Workshops*, pp. 369–374. Vienna, Austria, Springer.
128. Yang, H., Liu, J. (2021). Knowledge graph representation learning as groupoid: Unifying TransE, RotatE, QuatE, ComplEx. *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pp. 2311–2320. Association for Computing Machinery.
129. Wang, X., El-Gohary, N. (2023). Deep learning-based relation extraction and knowledge graph-based representation of construction safety requirements. *Automation in Construction, 147,* 104696.
130. Krompaß, D., Jiang, X., Nickel, M., Tresp, V. (2014). Probabilistic latent-factor database models. *Proceedings of the 1st Workshop on Linked Data for Knowledge Discovery Co-located with European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, pp. 74–83. Nancy, France, CEUR-WS.
131. Ji, H., Cassidy, T., Li, Q., Tamang, S. (2014). Tackling representation, annotation and classification challenges for temporal knowledge base population. *Knowledge and Information Systems, 41(3),* 611–646.
132. Peng, J., Lu, G., Shang, X. (2020). A survey of network representation learning methods for link prediction in biological network. *Current Pharmaceutical Design, 26(26),* 3076–3084.
133. Li, Z., Liu, H., Zhang, Z., Liu, T., Xiong, N. N. (2022). Learning knowledge graph embedding with heterogeneous relation attention networks. *IEEE Transactions on Neural Networks and Learning Systems, 33(8),* 3961–3973.
134. Zhang, Z., Li, Z., Liu, H., Xiong, N. N. (2022). Multi-scale dynamic convolutional network for knowledge graph embedding. *IEEE Transactions on Knowledge and Data Engineering, 34(5),* 2335–2347.
135. Wu, H., Song, C., Ge, Y., Ge, T. (2022). Link prediction on complex networks: An experimental survey. *Data Science and Engineering, 7(3),* 253–278.
136. Yang, S., Hu, B., Zhang, Z., Sun, W., Wang, Y. et al. (2021). Inductive link prediction with interactive structure learning on attributed graph. *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Database*, pp. 383–398. Bilbao, Spain, Springer.
137. Wang, Z., Zhang, B., Gao, D. (2022). A novel knowledge graph development for industry design: A case study on indirect coal liquefaction process. *Computers in Industry, 139,* 103647.
138. Yuan, J., Li, H. (2023). Research on the standardization model of data semantics in the knowledge graph construction of oil & gas industry. *Computer Standards & Interfaces, 84,* 103705.
139. Yin, Z., Shi, L., Yuan, Y., Tan, X., Xu, S. (2023). A study on a knowledge graph construction method of safety reports for process industries. *Processes, 11(1),* 146.
140. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. et al. (2022). Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, pp. 27730–27744. New Orleans, USA, Curran Associates, Inc.
141. Conover, M., Hayes, M., Mathur, A., Meng, X., Xie, J. et al. (2023). Free Dolly: Introducing the world's first truly open instruction-tuned LLM. https://www.databricks.com/blog/2023/04/12/dolly-first-open-commercially-viable-instruction-tuned-llm
142. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M. A. et al. (2023). LLaMA: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971.

143. Trajanoska, M., Stojanov, R., Trajanov, D. (2023). Enhancing knowledge graph construction using large language models. arXiv preprint arXiv:2305.04676.
144. Pan, S., Luo, L., Wang, Y., Chen, C., Wang, J. et al. (2023). Unifying large language models and knowledge graphs: A roadmap. arXiv preprint arXiv:2306.08302.