Article:
Wang, Yue, Wan, Yao, Bai, Lu et al. (2024) Collaborative Knowledge Graph Fusion by Exploiting the Open Corpus. IEEE Transactions on Knowledge and Data Engineering. pp. 475-489. ISSN: 1041-4347

https://doi.org/10.1109/TKDE.2023.3289949

Collaborative Knowledge Graph Fusion by Exploiting the Open Corpus

Yue Wang, Yao Wan, Lu Bai, Lixin Cui, Zhuo Xu, Ming Li
Philip S. Yu, *Fellow, IEEE* and Edwin R Hancock, *Fellow, IEEE*

Abstract—To ease the process of building Knowledge Graphs (KGs) from scratch, a cost-effective method is required to enrich a KG using the triples extracted from a corpus. However, it is challenging to enrich a KG with newly extracted triples since they contain noisy information. This paper proposes to refine a KG by leveraging information extracted from a corpus. In particular, we first formulate the task of building KGs as two coupled sub-tasks, namely join event extraction and knowledge graph fusion. We then propose a collaborative knowledge graph fusion framework, which is composed of an explorer and a supervisor, to allow the involved two sub-tasks to mutually assist each other in an alternative manner. More concretely, an explorer extracts triples from a corpus supervised by both the ground-truth annotation and the KG provided by the supervisor. Furthermore, a supervisor then evaluates the extracted triples and enriches the KG with those that are highly ranked. To implement this evaluation, we further propose a translated relation alignment scoring mechanism to align and translate the extracted triples to the KG. Experimental results verify that this collaboration can improve both the performance of our sub-tasks, and contribute to high-quality enriched knowledge graphs.

Index Terms—Knowledge Graph Enrichment, Joint Event Extraction, Knowledge Graph Fusion, Collaborative Learning

1 Introduction

Knowledge graphs, which are a structurally organized form of information, have supported a variety of downstream tasks, including recommender systems [1], NLP tasks [2], question answering [3], [4], and entity-linking [5]. Existing open-source knowledge graphs such as Wikidata [6], WordNet [7] and Freebase [8] comprise billions of Resource Description Framework (RDF) triples [9] in the form of *(subject, relation, object)* relations, where both the *subject* and *object* represent the named entities [10], and the *relation* describes the relationship between these two named entities. However, since open-source knowledge graphs are designed for general purposes, they contain only limited factual knowledge for particular tasks [11] in several domains such as finance or medicine. In order to effectively adapt to multiple domains, constructing high-quality domain-specific knowledge graphs is of utmost importance.

To construct new knowledge graphs from unstructured textual sources, current research mainly primarily involves several pipelined sub-tasks, e.g., named entity recognition [12], relation extraction [13] or relation alignment [14]. These methods are designed as separate sub-tasks rather than an integrated system [15]. Thus they do not fully address the issue of how to effectively leverage the information hidden in the connections between the sub-tasks [16] to improve the quality of a knowledge graph built from a text corpus. To this end, recent work has combined named entity recognition with relation extraction as a single joint-event-extraction [17] task that can jointly obtain the entities and relations from text sources. However, since the current work does not focus on the resulting process to build an integrated knowledge graph from the extracted results, there still exists much scope for constructing a high-quality domain-oriented knowledge graph from test documents.

Knowledge graph fusion [15], [18], [19] is a possible route by which to construct a knowledge graph from the extracted event factors in an open corpus. Early work applied the traditional data fusion method [20] while considering only fusing the data under a global or compatible data schema [21]. This work evaluates the quality of data by checking whether or not a triple is contained in the extended set of a ground-truth knowledge graph [22]. However, this type of method may ignore the implications of knowledge that is indirectly contained in the ground-truth knowledge graph. It may thus discard many meaningful triples from different and potentially valuable sources. In order to overcome this problem, recent knowledge graph embedding [23] methods have leveraged network embedding technology [24] to infer the possibilities of the existence of triples in a given knowledge graph. This is done by representing the triples as latent vectors [25], [26], [27]. Specifically, with the representation vectors of the triples to hand, these methods use statistical models [28] or neural

*Yue Wang, Lixin Cui, and Zhuo Xu are with the Central University of Finance and Economics, Beijing, China. Yao Wan is with the College of Computer Science and Technology at Huazhong University of Science and Technology (HUST), Wuhan, China. Lu Bai (*Corresponding Author: bailu@bnu.edu.cn; bailucs@cufe.edu.cn) is with the School of Artificial Intelligence, Beijing Normal University, Beijing, China, and the Central University of Finance and Economics, Beijing, China. Ming Li is with the Key Laboratory of Intelligent Education Technology and Application of Zhejiang Province, Zhejiang Normal University, Jinhua, China, and the Key Laboratory of Scientific and Engineering Computing (Ministry of Education), Shanghai Jiao Tong University, Shanghai, China. Philip S. Yu is with the Department of Computer Science, University of Illinois at Chicago, US. Edwin R. Hancock is with the Department of Computer Science, University of York, UK.* *This work is supported by the National Natural Science Foundation of China under Grants T2122020, 61976235, and 61602535. This work is also supported in part by NSF under grants III-1526499, III-1763325, III-1909323, and CNS-1930941, and the Program for Innovation Research in the Central University of Finance and Economics. Ming Li acknowledged the supports from the National Natural Science Foundation of China (No. 62172370, No. U21A20473), the Zhejiang Provincial Natural Science Foundation (No. LY22F020004), and the Fundamental Research Funds for the Central Universities.*

networks [29], [30] to predict plausible scores for the potential triples.

Challenges That Hinder the Emergence of a Unified Framework. Although many existing works have discussed the knowledge graph fusion task, few consider a unified framework that can automatically build a knowledge graph directly by absorbing a corpus. Hence, it is necessary to fuse the extracted triples from an open corpus to form a prior knowledge graph, or in other words, linking the candidate triple generation with the evaluation process. The main challenges that hinder progress in this direction are routed in the following shortcomings in the knowledge extraction and knowledge graph fusion tasks. (1) *Difficulties in aligning RDF triples*. Since open-text sources may contain relations outside the scope of a prior knowledge graph, it is a challenge to align the relations from the open texts to those in the knowledge graph. Although current work discusses the entity alignment [32] between sources, little attention has been paid to relation alignment. This leads to the difficulty of aligning the extracted RDF triples from the text sources to a prior knowledge graph. (2) *Difficulties maintaining knowledge graph quality*. Merging the unaligned RDF triples from the open text sources to a knowledge graph can mislead the knowledge graph embedding model and may result in unreliable plausible scores for potential triples. Moreover, a misleading knowledge graph can result in the extractor relying on low-quality triples. This may further lower the quality of the knowledge graph. (3) *Difficulties sharing knowledge between sub-tasks.* Without a reliable way of aligning the RDF triples, it becomes difficult to share knowledge between the sub-tasks (e.g. event extraction and knowledge fusion). This leads to error propagation [33] and thus degrades the performance for each sub-task.

To address the aforementioned limitations, in this paper, we formulate a new method that combines event extraction (extractor) with knowledge graph fusion as a Collaborative Knowledge Graph Fusion process. Specifically, we propose a unified framework to build a domain-oriented knowledge graph by enriching an open-source knowledge graph with knowledge extracted automatically from a text corpus. Since our new method provides a mechanism to share the knowledge between sub-tasks, our enriched knowledge graph grows larger by incorporating facts of knowledge from the texts. In addition, the new method also leverages the enriched knowledge graph to assist our event extraction sub-task to obtain more reliable entities and relations.

As illustrated in Figure 1, the collaborative knowledge graph fusion method consists of two interacting processes, an explorer and a supervisor. That is, by referring to the principles (e.g. the possible entity pairs) from a supervisor, an extractor explores new RDF triples from the available open text sources. After the extractor submits the newly discovered triples to the supervisor, the supervisor evaluates their quality and extends the existing set of triples using the highest quality newly discovered triples.

Specifically, our framework guides the extractor with the entity pairs from a prior seed knowledge graph, and then iteratively increments the seed knowledge graph with the extracted triples from the extractor. In this process, both the performance of the extractor and the quality of the enriched knowledge graph are improved. To this end, in our extractor, we propose a benchmark-based supervision mechanism to supervise the extraction process with the entity pairs from the seed knowledge graph maintained by the supervisor. This is implemented by a contrastive learning method which considers both the positive and negative entity pairs. These entity pairs are sampled from the prior knowledge graph with a neural knowledge-graph-embedding-based scoring function trained by the supervisor process. On the other hand, to the supervisor, the knowledge-graph-embedding-based scoring function is trained by the triples in the seed or the enriched knowledge graph and it evaluates the matching degree of the extracted RDF triples from the extractor to the knowledge of the supervisor. Consequently, the supervisor merges the high-ranked triples from the extracted results into the prior knowledge graph.

We conduct comprehensive experiments on real-world corpora and knowledge graphs. Experimental results show that our system achieves higher performance than state-of-the-art baselines, both on the joint-event-extraction and the knowledge-graph-embedding tasks. This verifies not only that the proposed benchmark-based supervision mechanism guides the extractor well in our system, and but that it also implies that the knowledge graph of the supervisor maintains high quality by being enriched with the triples evaluated by the supervisor.

Contributions. In summary, the primary contributions of this paper are as follows.

- We formalize the knowledge graph fusion with open corpora as an alternating process consisting of extracting the RDF triples from documents and then fusing a prior knowledge graph with the obtained triples. As far as we know, our work is the first to discuss a unified architecture to conduct the knowledge fusion directly based on the text sources.
- We propose the "Collaborative Knowledge Graph Fusion" framework as a solution for the aforementioned problem. In this framework, we propose the Benchmark-based Supervision Mechanism to further supervise the performance of our JEE process (in the explorer process) with positive and negative entity pairs sampled from a prior KG provided by the supervisor.
- We propose an unsupervised metric, Translated Relation Alignment Scoring (TRAS), to assist align and translate the extracted triples from the JEE process to those in the proper form to the prior KG.
- With the proposed Benchmark-based Supervision Mechanism and TRAS metric to hand, we implement the "Collaborative Knowledge Graph Fusion" as a unified process. It automatically extracts the triples from an open corpus and enriches them to a given prior KG in an alternative process.
- Our experiments on several real-world datasets show that, with the proposed framework, our system achieves better performance both on the JEE and KGF tasks than the related alternatives. This verifies that our method not only improves the JEE process but also yields a high-quality enriched KG. Specifically, our case study shows that our system could translate the extracted triples from a text corpus to

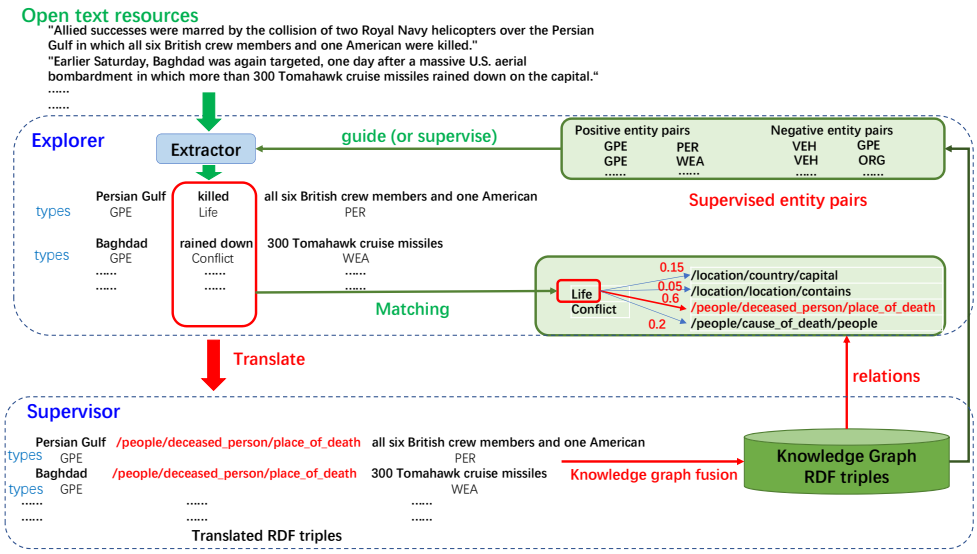


Fig. 1. In a collaborative knowledge graph fusion process, an explorer and a supervisor collaborate to create an enriched knowledge graph by extending a prior knowledge graph with RDF triples extracted from open text sources. Since the extracted RDF triples contain entities or relations that are not aligned to the prior knowledge graph, this process requires interaction mechanisms (translate the extracted results to the knowledge graph RDF triples and guide the explorer with meaningful entity pairs) between the explorer and the supervisor. To simplify the problem, we suppose both the explorer and supervisor share the same entity types (Geographical/Social/Political Entities (GPE), Persons (PER), Weapons (WEA), Organizations (ORG), Vehicles (VEH), etc.) and the extracted trigger text mentions (killed, rained down, etc.) by the explorer belong to the trigger types (Life, Conflict, etc.) by following the definitions in the ACE 2005 corpus [31]. Then the core problem becomes to align the trigger text mentions obtained by the explorer to the relations in the knowledge graph of the supervisor.

the facts consistent with a prior KG with the assistance of the proposed TRAS score. This improves the quality of the prior KG and also explains the reason for the performance improvement of the KGF task.

The remainder of this paper is organized as follows. In Section 2, we introduce the preliminaries concerning the joint event extraction and knowledge graph fusion processes and then also formalize the problem of knowledge graph fusion with an open corpus. Section 3 presents in detail our proposed framework and fusion mechanism. Section 4 verifies the effectiveness of our model and compares it with recent methods on real-world datasets. Section 5 summarizes recent related work. Finally, we conclude this paper in Section 6 where we offer suggestions for further work.

2 Preliminaries

Our overall objective is to fuse knowledge graphs by leveraging an open corpus. This task consists of a sub-task of Joint Event Extraction (JEE) to extract knowledge triples from unstructured texts and another sub-task of Knowledge Graph Fusion (KGF) to evaluate and enrich the extracted triples from the JEE for a prior or existing Knowledge Graph (KG). We first define some notations for the JEE and KG, and then formalize our problem in the following subsections.

2.1 Knowledge Graphs

A KG [34] is represented as a set of RDF triples referring to specific topics. Formally, we define a knowledge graph G as $G = \langle E, R, T \rangle$, where E is a set of entities, R is a set of relations and T is the set of the RDF triples. For example, $G_1 = \langle E_1, R_1, T_1 \rangle$ is a knowledge graph of capital city relationships with the entity set $E_1 = \{Tokyo, Beijing, Japan, China\}$, the relation set $R_1 = \{capital\_of\}$ and the triple set $T_1 = \{\langle Tokyo, capital\_of, Japan \rangle, \langle Beijing, capital\_of, China \rangle\}$. Since a human-composed document does not explicitly contain structural information, e.g., the entities, relationships, or triples, to build a KG from a corpus, we require to extract the triples from the texts.

2.2 Joint Event Extraction

Event extraction aims to extract structural information (e.g., entities or relations [12]) from a given corpus. It is typically composed of two sub-tasks of named entity recognition and relation extraction. Traditionally, separate multi-label classifiers are designed to predict the labels for both the entities and the mentioned relationship in a sentence. In order to improve the performance of event extraction, recent work resorts to pipeline-based methods, which first classify the relationship, and then identify the entities centered around the determined relation. However, since these methods perform the relation classification and entity identification sub-processes separately, these sub-processes hardly receive feedback from each other. As a result, those pipeline-based approaches may suffer from the error-propagation issue [35]. To this end, we put forward a universal sequence-to-sequence (Seq2Seq) framework [16] to simultaneously extract the entities and relations from a corpus.

Seq2Seq Joint-Event-Extraction (JEE). Let $\mathcal{D}$ be a corpus of textual sentences, where $\mathcal{D} = \{s_1, s_2, s_3, \ldots\}$. For each sentence $s \in \mathcal{D}$, $s = \{w_1, w_2, w_3, \ldots, w_m\}$, where w_i denotes a word token. Let $\mathcal{A} = \mathcal{A}_E \bigcup \mathcal{A}_R$ be a combined label set with predefined types for tokens, where $\mathcal{A}_E$ and $\mathcal{A}_R$ are the sets of the predefined entity and text relation mention types respectively. Then, the aim of JEE is to find

an optimal map $g_{\Theta_1} : s \rightarrow \Pi_{i=0}^{M}\mathcal{A}$, where Π is the Cartesian product, M is the maximum length for the sentences in $\mathcal{D}$, and Θ_1 denotes the learning parameters.

The loss function for JEE under the framework of Seq2Seq is designed as a cross-entropy function, as follows:

$$\mathcal{L}_{jee} = \sum_{i=0}^{M} \sum_{y_i \in \mathcal{A}} -Pr(y_i|w_i) \log \hat{Pr}(y_i|w_i) \,. \tag{1}$$

With the mapped label sequence optimized by the loss function in Eq. (1), we obtain the annotated label sequences for the sentences in a corpus. In this manner, the entity and relation text mentions for a sentence are extracted simultaneously. Consequently, we generate RDF triples based on their extracted text mentions and use these triples as the candidate triples for KG enrichment. For better illustration, we use the term g_{Θ_1} as a joint operation that combines both the mapping from sentences to label sequences and the RDF generation process. We refer to it as the *extractor map* in the following sections.

2.3 Knowledge Graph Fusion with an Open Corpus

Knowledge graph fusion [18] is the task of constructing a unified knowledge graph from different data sources. Traditional knowledge graph fusion aims to integrate several knowledge graphs into one knowledge graph, and we formalize this task as follows:

Knowledge Graph Fusion (KGF). Let $G_1 = \langle E_1, R_1, T_1 \rangle$ and $G_2 = \langle E_2, R_2, T_2 \rangle$ denote two prior knowledge graphs, where both G_1 and G_2 are used under the same RDF schema. $G' = \langle E', R', T' \rangle$ is the fused knowledge graph based on G_1 and G_2, where $T' = T_1 \bigcup \Delta T$ denotes the fused triple set which is based on T_1 and incremental triples ΔT from G_2 ($T' = T_1 \bigcup \Delta T$). The ΔT are the top-K triples that are close to G_1. This closeness is measured by the plausible score $f_{G_1}(i, r, t)$ ($(i, r, t) \in G_2$) which is computed as

$$f_{G_1}(i, r, t) = \sum_{(i^*, r^*, t^*) \in T_1} Sim((i, r, t), (i^*, r^*, t^*)) \,, \tag{2}$$

where Sim denotes the similarity between two triples.

We utilize a contrastive learning framework [36] to embed the triples as the corresponding vectors and implement the similarity between triple vectors via the translation-based embedding (TransE) [28] method.

Knowledge Graph Embedding (KGE). Given a KB $G = \langle E, R, T \rangle$, suppose (i, r, j) is a triple from T, we define the loss of knowledge graph embedding as follows:

$$\mathcal{L}_{kge} = - \sum_{\substack{(i,r,j) \in T, \\ (i',r,j') \in N}} ||\gamma + f_G(i, r, j)) - f_G(i', r, j')|| \,, \tag{3}$$

where N is the corresponding negative set for the triples in T, γ is a hyperparameter, and $f_G(i, r, j)$ is a scoring function to evaluate the consistency of any triple (i, r, j) to G. The normalization in Eq. (3) can be based on either the L1 or L2-norm. According to the design of TransE, a plausibility score $f_G(i, r, j)$ can be computed as follows.

$$f_G(i, r, j) = d(e_i + e_r, e_j) \,, \tag{4}$$

where e is an embedding that maps any entity or relation to an $\mathbb{R}^h$ vector, and $d(\cdot, \cdot)$ is the Euclidean distance function between two $\mathbb{R}^h$ vectors.

Therefore, with a trained embedding e based on the given prior knowledge graph G_1, the plausibility of a triple (i, r, j) from G_2 to G_1 can be evaluated by computing the Euclidean distance $d(e_i + e_r, e_j)$.

As discussed before, our objective is to fuse knowledge graphs by leveraging open text sources. This task is different from the aforementioned knowledge graph fusion, as we require to (1) extract the RDF triples from a given corpus $\mathcal{D}$ and (2) fuse the extracted triples to a knowledge graph G. Specifically, we formalize this problem as the following.

Open Knowledge Graph Fusion (OKGF). Given a prior knowledge graph $G = \langle E, R, T \rangle$, a corpus $\mathcal{D}$ and an extractor map g_{Θ_1}, suppose $g_{\Theta_1}(\mathcal{D})$ is a set of extracted triples from a corpus $\mathcal{D}$. Then with a trainable scoring function f and embedding map e, the objective of OKGF is to find the optimal subset ΔT from $g_{\Theta_1}(\mathcal{D})$ that minimizes the following loss function:

$$\mathcal{L}_{OKGF} = - \sum_{\substack{(i,r,j) \in T \bigcup \Delta T, \\ (i',r,j') \in N}} ||\gamma + f_G(i, r, j)) - f_G(i', r, j')|| \,, \tag{5}$$

where N is the corresponding negative triple set for the positive triples t from T.

This task combines the sub-tasks of JEE and KGF into a unified framework. However, it is a combinatorial optimization problem that exhaustively checks all the possible subsets ΔT from $g_{\Theta_1}(\mathcal{D})$. The newly discovered noisy entities and relations from the open corpus exacerbate the problem. Therefore, it is difficult to obtain the global optimal solution. To this end, we propose a heuristic collaborative knowledge graph fusion framework to connect the JEE and the KGF to fuse an open corpus into a prior knowledge graph. Our framework approaches the open knowledge graph fusion from two directions, namely 1) our model guides the JEE process with a prior knowledge graph, and 2) it selectively enriches the prior knowledge graph with the extracted results from the JEE process. This requires a careful design of both the JEE supervision mechanism with a knowledge graph and an effective "translate-and-evaluate" method to fuse the extracted results into the knowledge graph. We elaborate on the details in the next section.

3 Our Proposed Method

In this section, we introduce our proposed framework for collaborative knowledge graph fusion with an open corpus.

3.1 Overview

To emulate a human-like collaborative process for our task, we propose a framework with two components, namely 1) an explorer to explore the documents with JEE modules and 2) a supervisor to fuse the knowledge graph with the extracted results by the explorer. In the exploring process, we propose a benchmark-based supervision mechanism to assist the JEE task to extract the triples while guided by a supervisor (the benchmarks discovered by the supervisor from a prior KG). In the supervising process, we propose the Relation Alignment-based Knowledge Graph Fusion module to selectively accept the extracted triples to be added to the prior KG. These two components alternate to simultaneously extract knowledge triples and enrich a prior

KG with high quality. Figure 2 illustrates the architecture of our system. The details of the proposed processes are given in the following subsections.

3.2 The Explorer: Benchmark-based Supervision JEE

As shown in Figure 2, we perform the JEE in the exploring process. To ensure the explorer is guided by the supervisor, we introduce a Benchmark-based Supervision Layer to import the knowledge from the supervisor. In this work, we apply the Seq2Seq JEE as the basic extraction process and use BERT [37] as the encoder. This module can be substituted by any alternative JEE model if necessary.

Intuitively, during the exploratory period, an explorer receives examples from a supervisor and attempts to leverage the knowledge in these examples to facilitate better exploration. In our work, the explorer extracts the triples from an open corpus based on a prior KG maintained by a supervisor. Since the open corpus may contain unaligned relations and extra entities that are not contained in the prior KG, it requires a relatively flexible method rather than strict supervision to guide the explorer. To this end, we introduce the benchmark-based supervision mechanism. The benchmark here means the supervisor-provided target that the explorer tries to reach.

Benchmark-based Supervision Mechanism. Given a prior KG, $G = \langle E, R, T \rangle$, let a positive set of entity pairs P^+ and a negative set of entity pairs P^- be a benchmark, where $P^+ = \{(i,j)|(i,*,j) \in T, \forall i,j \in E\}$, and $P^- = \{(i,j)|(i,*,j) \notin T, \forall i,j \in E\}$. Then, the Benchmark-based Supervision Mechanism can be described as the task to minimize a loss function extended from the Bayesian Personalized Ranking (BPR) loss [38], as follows:

$$\mathcal{L}_b = -\log\left(\delta(f(P^+) - f(P^-)\right), \tag{6}$$

where δ is the Sigmoid function. Function $f(P)$ computes the likelihood for any entity pair $(i,j) \in P$, given by

$$f(P) = \mathrm{ffnn}(\sum_{\mathrm{(i,j)\in P}} (\mathrm{e_i} - \mathrm{e_j})), \tag{7}$$

where P is the set of all the related pairs ($P = P^+ \bigcup P^-$); e_i is an $\mathbb{R}^d$ embedding vector for any entity i (where $i \in E$); "ffnn" is a Feed-Forward Neural Network layer to map an $\mathbb{R}^d$ embedding vector to an $\mathbb{R}^1$ score.

Optimizing $\mathcal{L}_b$ results in the training of a scoring function $f(P)$ to measure the likelihood of any entity pair while maximizing the difference between the likelihood scores of the positive and negative entity pairs. This fits with the intuition that an explorer understands the knowledge in the examples from the supervisor.

Furthermore, since an entity is a sequence of tokens with arbitrary lengths, we apply the weighted average method [39] to represent an entity by its corresponding embedding vector. Formally, the embedding vector for an entity is computed as follows:

$$e_i = \sum_{w \in i} e_w\,, \tag{8}$$

where i is an entity in E and w is any token in the entity i. The embedding vector e_w can be obtained by referring to the embedding dictionary table.

With the proposed Benchmark-based Supervision Mechanism, the loss function of our explorer process is a weighted sum of the losses in Eq. (1) and Eq. (6), as follows:

$$\mathcal{L}_e = (1-\alpha)\mathcal{L}_{jee} + \alpha\mathcal{L}_b\,, \tag{9}$$

where α is the weight for the benchmark-based supervision. The benchmark-based supervision loss $\mathcal{L}_b$ in $\mathcal{L}_e$ guides the explorer to extract the conformed event factors based on the examples from the supervisor. These conformed factors are also crucial to improve the quality of the knowledge graph of the supervisor. In our experiments, both our explorer and supervisor perform the best when $\alpha = 0.5$.

Candidate Triple Set. With the aforementioned explorer process, our system simultaneously extracts the entity and relation text mentions (or triggers). Then, we generate all RDF triples exhaustively based on the extracted text mentions. The results are treated as the candidate triple set T' for subsequent processing steps.

3.3 The Supervisor: Relation Alignment-based OKGF

Our supervisor process enriches the prior KG with the optimal subset of candidate triples from the explorer process. This requires a scoring function to measure the plausibilities for triples trained by the prior KG. The process for a supervisor to evaluate the quality of the discovery from the explorer is similar to that adopted by the explorer. As discussed in Section 2.3, one of the challenges to implementing this task is that the relation text mentions from the candidate triples may not be unaligned to the relations in the prior KG. In order to address this issue, we propose the Translated Relation Alignment Score (TRAS). This score facilitates the alignment of the relations between the candidate triples and the existing relations in the prior KG. After aligning the relations, our system translates the candidate triples to the aligned candidate triples. It then ranks the aligned candidate triples by considering the semantic information residing in the prior KG. The highly-ranked triples are integrated into the prior KG to generate an enriched KG. We expand the details of this process in the remainder of this section.

Translated Relation Alignment Score (TRAS). Given two KGs $G_1 = \langle E_1, R_1, T_1 \rangle$ and $G_2 = \langle E_2, R_2, T_2 \rangle$ ($T_1 \bigcap T2 = \phi$). The TRAS score $s(r_1, r_2)$ between two relation $r_1 \in R_1$ and $r_2 \in R_2$ is computed as follows:

$$s(r_1, r_2) = \gamma Sim_m(r_1, r_2) + (1-\gamma)Sim_e(r_1, r_2)\,, \tag{10}$$

where $Sim_m(r_1, r_2)$ is the text mention similarity between r_1 and r_2, γ is the weight of the text mention similarity. The quantity $Sim_e(r_1, r_2)$ is the **translated relation similarity** between two relations (i.e., r_1 and r_2), which can be computed as follows:

$$Sim_e(r_1, r_2) = Sim(\sum_{(i,r_1,j)\in T_1} e_i - e_j, \sum_{(i,r_2,j)\in T_2} e_i - e_j)\,, \tag{11}$$

where $Sim(\cdot,\cdot)$ can be any similarity function between two vectors. In this paper, we use Cosine similarity for this task. The summed entity embedding difference in Eq. (11) represents the proximity between two relations in different KGs. Generally, a larger value of γ gives a greater weight to the text mention similarity. A smaller value of γ allows our model to capture more indirect semantic information

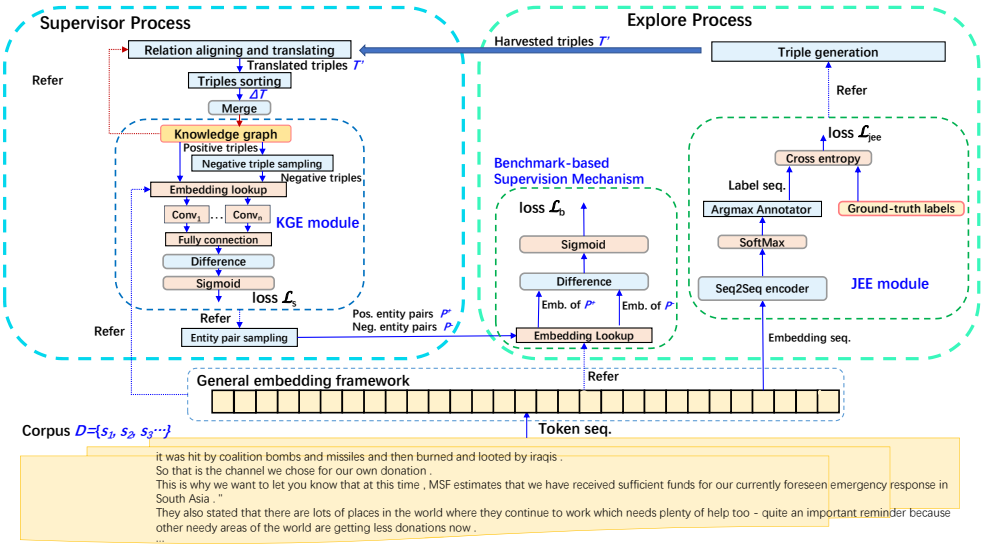


Fig. 2. An overview of the collaborative knowledge graph fusion framework via leveraging open corpus, which consists of two alternative running processes: 1) an exploring process carries on the Joint-Event-Extraction (JEE) task and 2) a supervising process aligns and merges the extracted triples to a prior knowledge graph. Our framework first embeds the texts to the latent vectors of tokens and then optimizes the forward scores for the explorer process. After training the JEE model, our system extracts the triples T' from the open texts. Then, our system treats them as candidate triples and enriches them to the prior KG by referring to the proposed Translate Relation Alignment Score (TRAS). The enriched KG and the trained KGE likelihood scoring function help to sample the top positive and negative entity pairs for the explorer process in return.

around relations. In our experiment, our supervisor performs the best when $\gamma = 0.5$.

Aligned Triple Set. Our system ranks the relation pairs between the candidate triples from T' and the triples in the prior KG using their TRAS scores. As a result, our system translates the candidate triples from the JEE process to an aligned triple set with the same relation set in the prior KG. The aligned triple set is denoted by ΔT.

Knowledge Graph Embedding (KGE) Triple Likelihood. After generating the aligned candidate triple set from the extracted triples, the supervisor ranks the candidate triples and merges the top-ranked triples to the current prior KG. To this end, we use a Knowledge Graph Embedding (KGE) Triple likelihood to perform the ranking task for triples. This function represents the action of the supervisor and it is implemented using a Convolutional Neural Network (CNN) [40] based model to map the triples to an $\mathbb{R}^1$ score. Theoretically, our framework can enhance the performance of the supervisor with any KGE module. The CNN-based KGE is a commonly used method [41], [42] in recent KGE works, since it obtains the latent features automatically. This module can be substituted into consequential works if necessary.

Formally, given a KG $G = \langle E, R, T \rangle$. For any two entities i and j ($i \in E$ and $j \in E$) and a relation $r \in R$, the KGE triple likelihood $f_G(i, r, j)$ is computed as follows:

$$f_G(i, r, j) = \delta(F([C_1, C_2, C_3, \ldots, C_m])), \tag{12}$$

where δ is the Sigmoid function, F is a fully-connected layer to map the concatenated convolution results to a $\mathbb{R}^1$ score that refers to the plausible probability for the triple (i, r, j) based on G. The quantity C_n is the n-th convolutional result which can be computed as follows:

$$C_n = Maxpool(Relu(W_n \otimes [e_i^T, e_r^T, e_j^T] + B_n)), \tag{13}$$

where W_n is the n-th (n=1, 2, ..., m) convolutional kernel and B_n is the corresponding bias, $\otimes$ is the convolution operator, $Maxpool$ is the Maxpooling function, $Relu$ is the ReLU active function and e_r is the embedding vector for the relation r. To alleviate the problems of sparsity in the extracted relations, rather than the one-hot encoding with a fixed dictionary, we applied a similar method to Eq. (8) to sum all the tokens in a relation mention to obtain the embedding vector e_r of a relation r.

The KGE triple likelihood is trained by optimizing a BPR loss function

$$\mathcal{L}_s = - \sum_{\substack{(i,r,j) \in T \bigcup \Delta T, \\ (i',r,j') \in N}} \log \left[\delta(f_G(i, r, j) - f_G(i', r, j')) \right]. \tag{14}$$

Optimizing $\mathcal{L}_s$ maximizes the difference between the positive and negative triples. Since this training uses all of the triples in G, the trained KGE triples likelihood represents the action of a supervisor based on the current KG.

Benchmark Entity Pairs Sampling. With the KGE triple likelihood to hand, we propose an algorithm (cf. in Algorithm 1) to obtain the top positive and negative set pairs based on the current KG and embedding.

The sampled positive and negative entity pairs are used directly as the benchmarks to supervise the explorer process (cf. Eq. (6)). This simulates the way in which the supervisor provides the key examples to the explorer for the exploration task.

3.4 The Complete Process and Discussion

The complete Collaborative Knowledge Graph Fusion process is described in the Algorithm 2. We initialize the embeddings for all tokens in the corpus with pre-trained features (BERT [37] in this paper, but alternative methods could potentially be used if necessary). These embeddings are then used in the supervisor process to infer the positive

Algorithm 1: Benchmark Entity Pairs Sampling

Data: a KG $G = \langle E, R, T \rangle$, the embedding mapper $\mathcal{E}$ from the JEE process, a threshold k.
Result: the positive entity pair set P^+, the negative entity pair set P^-.

begin
  Compute all $f_G(i, r, j)$s (where $(i, r, j) \in T$) with Eq. (12).
  Sort the triples in T in ascending order and select the top-k ranked entity pairs P^+.
  Enumerate all the negative triples N (where $(i, r, j) \notin T, i, j \in E, r \in R$).
  Compute all $f_G(i, r, j)$s (where $(i, r, j) \in N$) with Eq. (12).
  Sort the triples in T' in descending order and select the top-k ranked entity pairs P^+.
  Output P^+ and P^-.
end

Algorithm 2: Collaborative Knowledge Graph Fusion Algorithm

Data: A prior KG $G = \langle E, R, T \rangle$, a corpus $\mathcal{D}$ and a threshold k for the polarity triple sampling and a threshold ε for the KG enrichment.
Result: An enriched KG G'.

1 **begin**
2 Initialize the embedding mapper $\mathcal{E}$ for all the tokens using the pre-trained features.
3 let $G' \leftarrow G$, $T' \leftarrow \phi$.
4 **while** *Round in* $[0, K)$ **do**
5 **Supervisor process:**
6 **if** $T' \neq \phi$ **then**
7 Align the relations in T' to R with Eq. (10).
8 $\Delta T \leftarrow$Find the top-K triples in the aligned T' with the trained $f_{G'}(*)$.
9 $T' \leftarrow \Delta T \bigcup T'$.
10 **end**
11 Sample the negative triple set N based on T'.
12 Train the KGE triple likelihood $f_{G'}(*)$ by minimizing Eq. (14), with T' and N.
13 Sample the top-k positive and negative entity pairs P^+ and P^- based on Algorithm 1 with T' and the embedding map $\mathcal{E}$.
14 **Explorer process:**
15 Train the benchmark-based supervision JEE by minimizing the function in Eq. (9) with JEE training data.
16 Exhaustive generate the candidate triples T' based on the mention results from the JEE testing data with the trained JEE.
17 **end**
18 **Output** G'.
19 **end**

or negative entity pair sets using a prior knowledge graph. Next, the obtained positive and negative entity pair sets are used to supervise the explorer process. Then, the JEE model in the explorer process extracts improved entities and relations to enrich the prior knowledge graph. The supervisor adds the top-K ranked aligned candidate triples in using beam search.

Discussion and Analysis. Our model links event extraction and knowledge graph fusion together as a single process. This alternative process enhances the performance of both of the aforementioned tasks and also results in a high-quality enriched KG. The main reasons for these improvements are twofold. First, with more useful knowledge implications (evaluated extracted triples from the corpus) for a given knowledge graph, the semantic relationships between its entities are improved. As a result, the performance of the knowledge graph embedding with the enriched knowledge graph is also improved. Second, the accuracy for the entity and relation extraction tasks is also improved with the help of the enriched knowledge graph.

3.5 Negative Triple Sampling and Training

Many existing methods use the randomized head or tail entity to replace triples from the positive triple set as the negative samples [43]. To further improve the quality of the negative samples in Line 11 of Algorithm 2, we treat the output of random sampled negative triples as the candidate set and then further use the KGE triple likelihood to measure their likelihoods. The final negative samples set in Line 11 of Algorithm 2 are the top-ranked samples from the candidate set based on the KGE triple likelihood scores.

4 Experiments and Analysis

In this section, we aim to address the following research questions:

- **RQ1:** Can a system in the proposed Collaborative Knowledge Graph Fusion framework successfully improve the performances for both the JEE and KGF (or KGE) tasks?
- **RQ2:** What is the generalizability of the proposed Collaborative Knowledge Graph Fusion framework representation across different real-world corpora and KGs?
- **RQ3:** Do the automatically extracted and translated triples represent valuable additional knowledge for the target KG?

We also perform an ablation analysis to investigate the effect of each module of the model in turn, as well as a qualitative analysis of detailed examples.

4.1 Datasets

Since our system consists of the optimization processes of the JEE together with the KGF, our dataset contains several real-world corpora to test the JEE and also two public KGs to test the KGF.

The Corpora. ACE 2005 [31] is a widely used dataset that has been adopted to test the performances of event extraction models. WebNLG is a corpus used for a challenge involving natural language generation [44]. CoNLL is a Spanish news corpus from [45]. We create the NYT and CoNLL datasets[1] used in our study by preprocessing the original NYT [46] and CoNLL [45] corpora with the CoreNLP[2]. This preprocessing includes annotating the triggers and entities from the text sentences.

The Knowledge Graphs. In order to implement the benchmark-based supervision mechanism in the explorer

1. https://github.com/hkharryking/labeled_NYT_CoNLL
2. https://stanfordnlp.github.io/CoreNLP/

process, we preprocess WN18 and FB15k-237 [28] and use them as the prior KGs in our evaluation. Since the entities in each KG are encoded as inner IDs, we map these IDs to real entity text mentions using the corresponding mapping files. Further, since the freebase API is deprecated, we map the entity IDs in FB15k-237 to the URLs on Wikidata.[3]) and then crawl the Wikidata titles to create the real entity text mentions.

Preprocessing Details. To implement a complete "Collaborative Knowledge Graph Fusion" framework, we preprocess the datasets to obtain training sets and testing sets respectively for the supervisor and explorer. The details of these preprocessed datasets are listed in Tables 1 and 2.

TABLE 1
Summary of the Corpora for the Explorer (JEE) Process

	ACE2005	NYT	CoNLL	WebNLG
Sentences	17,606	6,355	3,903	3,973
Training sent.	16,765	5,500	3,000	2,649
Testing sent.	841	855	903	1,324

TABLE 2
Summary of the KGs for the Supervisor (KGF) Process

		ACE2005	CoNLL	NYT	WebNLG
FB15K	Seed triples	20,00	3,440	3,000	3,973
	Testing triples	969	698	1,129	1,786
WN18	Seed triples	526	68	2,042	311
	Testing triples	129	68	730	113

4.2 Comparison Baselines

To evaluate the performance of our proposed approach, we compare it with the following baselines for both the JEE and the KGF tasks.

JEE Baselines

- StagedMaxEnt [35] and TwoStageBeam [47] are two classic methods based on a pipe-lined framework to jointly extract the event factors.
- Reranking [35] is a statistical state-of-the-art joint event extraction method.
- Seq2Seq [48] is a model for JEE based on the sequence-to-sequence framework. Our experiments use the universal sequence-to-sequence framework implementation from [16].
- Seq2Seq* [48] is the extended Seq2Seq model with the Glove [49] pre-trained features.
- CRF* [48] is a method extended from Seq2Seq with a conditional random field layer containing the Glove [49] pre-trained features.
- BERT [37] is the original BERT with Seq2Seq downstream layers.
- Joint3EE [50] is an embedding-based method to extract the entities, event triggers and arguments together.
- REKnow [51] is a Seq2Seq joint method that leverages knowledge bases to obtain enhanced features.
- Benchmark-based Supervision JEE (BJEE) is the joint model proposed in this paper. Our model is supervised using the benchmark entity pairs sampled from a given knowledge graph. It is based on the explorer process described in Sec. 3.2. The subscripts in the experimental results are the names of the given knowledge graphs.

KGF Baselines

- TransE [28] is a classic statistical KGF model. It assumes that the triple relations can be represented as the difference between the head and tail entity vectors of the triples. The method trains the latent vectors for all the triples based on this assumption.
- ConvE [29] is a KGF method that concatenates the vectors for entities to create a matrix to represent the triples. It applies a convolutional neural network to capture the proximity between entities in a triple.
- Supervisor is the method proposed in this paper and described in Sec. 3.3. It iteratively enriches its training knowledge triples with the results extracted from the explorer process.

4.3 Evaluation Metrics

Evaluation metrics for JEE. The performance of JEE is measured by the Precision, Recall, and F1-scores for the triggers, the entities, and the arguments. Precision is measured by the ratio of the number of correct tags output from all the tokens in a corpus. Recall is the ratio of the number of predefined tags contained in the output tags.

Evaluation metrics for KGF. In the KGF task, we use MRR, Hit@30, Hit@40, and Hit@50 as the metrics to measure how well a model predicts the possibility of a triple. Concretely, the MRR (Mean Reciprocal Rank, MRR) is computed using the definition:

$$MRR = \sum_{t \in \hat{T}} \frac{1}{rank_t}, \tag{15}$$

where $\hat{T}$ is the test triple set and $rank_t$ is the practical rank for t in the predicted list. Hit@n is the ratio of the number of positive triples that are in the top-n ranked triples ($n = 30, 40, 50$ in our experiment) in the test triple set $\hat{T}$.

Since our method runs using the JEE and KGF tasks alternately, in order to improve efficiency, we pre-sampled the positive and negative triples from the test triple set and saved them to files. Our evaluation of the performances of the KGF tasks is based on these pre-sampled triples.

4.4 Prototype System and Implementation Details

We implement a prototype system with the proposed Collaborative Knowledge Graph Fusion framework with PyTorch. This system consists of a) an explorer process that performs the JEE task to extract the triples from a corpus and b) a supervisor process that conducts KGF to train the KGE triple likelihood based on the prior KG. As introduced in Section 3, our system enriches a prior KG as follows. Initially, the explorer process extracts the triples from a given corpus under the guidance (the Benchmark-based Supervision Mechanism) of the supervisor. After the explorer submits the triples to the supervisor, the supervisor translates the triples to match the form of its prior KG. With the translated triples, the supervisor assesses their quality based on the KGE triple likelihood (which represents its

3. https://www.wikidata.org

TABLE 3
Detailed comparison on ACE 2005 testing set.

Model	Event Trigger Identification			Event Trigger Classification			Event Argument Identification			Event Argument Classification		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
StagedMaxEnt	73.9	66.5	70.0	70.4	63.3	66.7	75.7	20.2	31.9	71.2	19.0	30.0
TwoStageBeam	76.6	58.7	66.5	74.0	56.7	64.2	74.6	25.5	38.0	68.8	23.5	35.0
Reranking	77.6	65.4	71.0	**75.1**	63.3	68.7	73.7	38.5	50.6	70.6	36.9	48.4
Joint3EE	70.5	74.5	72.5	68.0	71.8	69.8	59.9	59.8	59.9	52.1	52.1	52.1
Seq2Seq	66.7	62.4	64.5	57.3	53.7	55.5	62.8	72.8	67.5	46.3	56.6	50.9
Seq2Seq*	72.4	67.5	69.9	69.7	65.0	67.2	72.7	75.0	73.8	58.7	67.0	62.6
CRF*	71.9	73.6	72.7	68.2	68.2	68.2	70.7	**79.6**	74.9	58.7	66.0	62.1
BERT	75.0	**75.0**	75.0	75.0	**75.0**	75.0	82.8	72.6	77.4	71.4	69.0	70.2
BJEE$_{wn18}$	88.9	66.7	76.2	85.7	60.0	70.6	88.2	77.8	82.7	80.4	72.6	76.3
BJEE$_{fb15k}$	**88.9**	72.7	**80.0**	**88.9**	72.7	**80.0**	**89.0**	77.7	**83.0**	**86.5**	**69.8**	**77.2**

TABLE 4
Comparison on the entity extraction on the ACE2005 testing set.

Model	Precision	Recall	F1
Seq2Seq	67.5	83.2	74.6
Seq2Seq*	74.4	85.1	79.4
CRF*	75.2	84.6	79.6
Reranking	82.4	79.2	80.7
PipelineGRU	80.6	80.3	80.4
Joint3EE	82.0	80.4	81.2
BERT	89.2	78.3	83.4
BJEE$_{wn18}$	92.4	81.5	86.6
BJEE$_{fb15k}$	**95.1**	**83.0**	**88.6**

own understanding of the prior KG). Finally, the supervisor merges high-quality triples found in the previous step, adding them to its prior KG, and then also updates the benchmarks used by the explorer.

For fair comparisons, all of the sequence-to-sequence encoders were implemented based on a BERT [37] with 768 hidden dimensions. Since our framework requires two alternating processes, we use an Adam optimizer [52] with a $1e$-3 learning rate and 30 epochs to train the explorer process for non-BERT models and all of the BERT-based models (including our own) are trained with a $2e$-5 learning rate and 30 epochs. We apply an Adadelta [53] optimizer with a $1e$-1 learning rate and 20 epochs to train the supervisor process. The number of rounds performed by the Collaborative Knowledge Graph Fusion framework is set to 8 for all of our models. Both the weights for the benchmark-based supervision and the mention similarity (α and γ) are set to 0.5 in the prototype system. The prototype system runs on a Linux machine with 4 NVIDIA 2080TI GPUs.

4.5 Comparison on JEE (RQ1 and RQ2)

We compare our model with the alternatives on the standard event extraction dataset ACE 2005. The results of the event trigger and argument extractions are shown in Table 3. The performances on all related sub-tasks of our model are superior to the alternatives. We further compare the performance of the text entity detection of our model with the alternative methods. Here our method also outperforms the alternatives (in Table 4). All of these results verify the effectiveness of the proposed supervisor-explorer mechanism in improving the performance of the JEE process. We also find that the sequence-to-sequence (Seq2Seq) uniform framework improves performance on the argument identification and classification tasks.

To validate the universality of our method, we compare the overall extraction performance for the proposed JEE models guided by FB15K and WN18 knowledge graphs on each of the real-world datasets in Table 5. Since many of the published methods do not report results on these datasets, we only report the results of our implemented methods in this experiment. Our proposed method extracts better text mentions (both the event argument and trigger mentions) than the alternative non-knowledge-base-guided methods. Furthermore, an interesting observation that can be drawn from these results is that, although the CONLL is a Spanish corpus, the performances of the event extraction tasks on it can still be improved by the proposed framework with the English-text knowledge graphs (FB15K and WN18). The reason for this is that many proper nouns are shared by both Spanish and English, and their semantic structure may assist the event extraction in Spanish. All of the results in this experiment verify that the proposed collaborative knowledge graph fusion framework effectively improves the performance of the JEE processes.

4.6 Comparison on KGF (RQ1 and R2)

In this experiment, we compare the performance of our method with the alternative KGF models on the triple prediction task. The experiment is conducted in the following way. First, the classic models TransE and ConvE are directly trained on the training set of the knowledge graph FB15K. The supervisor of our model is trained with an enriched training set that is obtained through the proposed Supervisor-explorer Collaborative Learning process. Second, all of the models are tested with the same testing set of FB15K. The results of the supervisor model are obtained by alternately running the supervisor and explorer processes for 8 rounds. Third, since exhaustively enumerating all of the negative triples requires weeks of computing time on our hardware platform, we only used 200 sampled negative triples with the corresponding positive triples as the test set when computing the performance metrics. The results of this experiment are listed in Table 7. From Table 7, we observe that with the enriched triples, the performance of our KGF model is improved. This verifies that the obtained triples from our Collaborative Knowledge Graph Fusion framework bring useful information to predict the potential knowledge triples in a knowledge graph and the quality of the seed knowledge graph is enhanced.

4.7 Ablation Analysis (RQ2)

Since we use BERT [37] as the sequence-to-sequence encoder for our model, we compare the experimental results of our models (BJEE$_{wn18}$ and BJEE$_{fb15k}$) with the pure BERT [37]

TABLE 5
Comparison on all the real-world datasets with overall performances.

Model	ACE 2005			NYT			CoNLL			WebNLG		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Seq2Seq*	71.2	73.9	72.5	91.0	88.2	89.5	86.6	88.7	87.6	91.2	90.9	91.1
CRF*	71.3	76.5	73.8	89.9	89.8	89.9	87.3	88.6	88.0	92.2	89.6	90.9
REKnow	71.3	67.6	69.4	93.1	94.1	93.6	-	-	-	90.4	87.9	89.1
BERT	87.1	86.9	87.0	97.8	97.8	97.8	94.2	94.2	94.2	90.1	90.0	90.1
BJEE$_{fb15k}$	92.1	92.1	92.1	**99.2**	**99.2**	**99.2**	**96.3**	**96.3**	**96.3**	96.3	96.3	96.3
BJEE$_{wn18}$	**96.0**	**94.9**	**95.5**	99.0	99.0	99.0	95.8	95.6	95.7	**98.2**	**98.2**	**98.2**

TABLE 6
Top extracted and aligned results from ACE 2005 corpus to knowledge graph FB15k by our system.

Rank	ACE 2005 corpus				FB15K
	Head Entity	Trigger mention	Trigger type	Tail Entity	Relation
1	the Persian Gulf	killed	Life	all six British crew members	/people/deceased_person/place_of_death
2	two Royal Navy helicopters	killed	Life	all six British crew members and one American	/people/cause_of_death/people
3	the capital	rained down	Conflict	aerial more than 300 Tomahawk cruise missiles	/people/deceased_person/place_of_death
4	the United States	summit	Contact	the president Putin	/business/business_operation/industry
5	the capital	took control	Baghdad the police stations	Movement	/location/country/form_of_government

TABLE 7
Comparison on the KGF task on the FB15K.

Model	Precision	Hit@30	Hit@40	Hit@50	MRR
TransE	62.5	14.5	18.5	23.0	0.0219
ConvE	84.0	15.0	20.0	25.0	0.0281
Supervisor	**98.5**	**15.0**	**20.0**	**25.0**	**0.0294**

model (with the same hidden dimensions) in Table 3, Table 4 and Table 5. With the proposed benchmark-based supervision mechanism, our results significantly outperform those obtained with pure BERT after the iterative learning process between the supervisor and explorer. To further discuss the influence of the iterative process, we also provide an experiment to compare the overall JEE performances with different iterative rounds. The results are shown in Figure 3. From this figure, the overall JEE performance improves with an increasing number of iterations. This shows that the alternating iterative process between the explorer and supervisor in our model improves the overall performance of the JEE task.

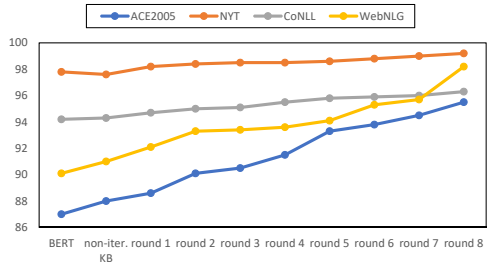


Fig. 3. The overall extraction performance of the explorer process with different rounds and supervised under the WN18 knowledge base

4.8 Sensitivity Analysis

In order to further analyze the details of the proposed Collaborative Knowledge Graph Fusion framework, we provide several experiments to study its performance with different forms of the teacher and explorer processes.

Figure 4 shows the performance of our system with a fixed teacher (with 4 CNN kernels) and explorers with different numbers of hidden dimensions. From the figure, it is clear that with the same teacher, then an explorer with more hidden dimensions performs better. Figure 5 gives

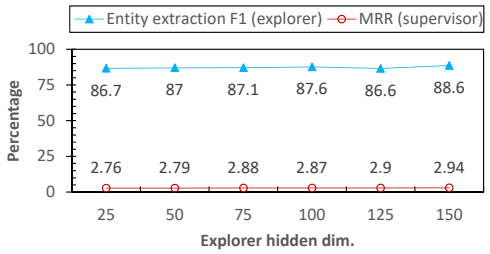


Fig. 4. The performances of our system under different explorers.

the performance of our system with a fixed explorer (with 150 hidden dimensions) under supervisors with different numbers of CNN kernels. From this figure, we observe that, with the same explorer, the performance of our system is optimal for a particular choice of the number of CNN supervisor kernels. In this experiment, the optimal number of kernels is 32. The two aforementioned experiments in-

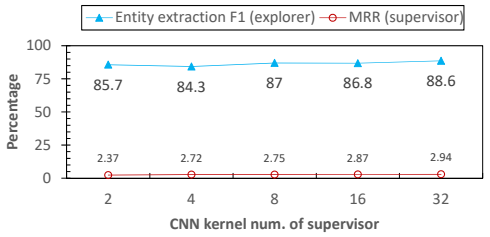


Fig. 5. The performances of our system under different supervisors.

dicate that the overall performance of a system with the proposed framework can be optimized by improving the explorer process, and the overall performance improvement is limited by the explorer under different supervisors.

4.9 Case study: Translate and Align the Triples (RQ3)

As introduced in Algorithm 2, the explorer process of our system extracts new triples from the given corpus (ACE 2005) and generates a mapper to align the relations of these extracted triples to the relations in the knowledge graph (FB15K). Then, with the aligned relation mapper, our

prototype system translates all of the extracted triples into the form of the target knowledge graph. In the final step, the explorer process ranks these translated triples with the trained KGE likelihood function from the supervisor and submits the top-ranked triples to the supervisor.

To further analyze the detailed performance of the proposed TRAS (Translated Relation Alignment Score) method, we explore the automatically aligned relations by our Collaborative Knowledge Graph Fusion framework in the task to explore (extract) the ACE 2005 corpus guided by the FB15K knowledge graph.

We select some top-ranked aligned and translated triples from the ACE 2005 corpus identified by our system and list them in Table 6. Most of these triples are aligned to the appropriate relations in FB15K and thus generate proper triples for FB15K based on the given corpus. For example, our system aligns and the trigger mention "killed" of the type "Life" to the FB15K relation "/people/deceased_person/place_of_death" for the 1-st triple extracted from the ACE 2005 corpus. Our system infers that the trigger mention "killed" of the ACE 2005 corpus is highly similar to the relation "/people/deceased_person/place_of_death" of the knowledge graph FB15K. In this result, our system infers that the trigger mention "killed" of the ACE 2005 corpus is aligned to the relation "/people/deceased_person/place_of_death" of the knowledge graph FB15K. Our system makes this inference by considering both the semantic similarity between the text mentions 'killed" and "deceased" and the affinities of the "PER" entities around the corresponding relations in the two sources. This shows that the proposed TRAS score provides a possible route for fully-automatic knowledge graph fusion in future work.

5 Related Work

5.1 Joint Event Extraction

Joint event extraction (JEE) [54] aims to simultaneously obtain the named entities, trigger text mentions, and relations from a given corpus. Much recent work applies the pipe-lined method to achieve this goal. This is a two-step process. First, they train a series of classifiers for the aforementioned sub-tasks and classify the text mentions in sentences as different triggers. Second, the classified triggers are used to identify the entity text mentions or relations. StagedMaxEnt [35] and TwoStageBeam [47] are examples of such pipe-lined systems. Reranking [35] is a state-of-the-art statistical pipe-lined method for the JEE task.

Most neural network models apply the embedding method to capture the latent semantic relationships between sentence tokens and attempt to train different classifiers for different sub-tasks. Joint3EE [50] is such a method that uses the multitask learning framework. However, since the separate training required for different classifiers increases the sparsity of the samples needed for each individual classifier, the performance improvement from these methods is limited. Recent work [55] provides end-to-end models for this task. Sequence-to-sequence methods [16] train a neural network to match a sentence in the form of a token sequence to a labeled sequence. This type of method reduces all of the individual sub-tasks to a single classifier and alleviates the sparse problem of entity relationships. Moreover, RE-Know [51] leverages knowledge bases to obtain enhanced features for the entities, and thus further improves the performance of the sequence-to-sequence joint method.

5.2 Knowledge Graph Fusion

Knowledge graph fusion [18] aims to fuse a knowledge graph with additional data sources. Many KGF systems apply an "enumerate-and-rank" framework [26] to complete the knowledge graph. That is, they train classifiers based on a given knowledge graph and identify the possible triples from a series of candidate triples. Usually, such classifiers are based on the knowledge graph embedding (KGE) [56] method. TransE [28] is a classic KGE method used to learn the embedding vectors needed to represent the triples in a knowledge graph. Much recent work applies neural network methods to improve the performance of the KGE task. ConvE [29] is a neural network KGE model with convolutional neural network modules. Recent work focuses on providing the embeddings by considering the heterogeneity of the knowledge graphs [57] or the heterogeneous information networks [58]. However, to the best of our knowledge, none of the existing methods directly considers the link the JEE to the KGE task.

5.3 Open Information Extraction

Open Information Extraction (Open IE) [59] is an alternative way to generate structural information from text sources. Traditional methods [60] obtain new facts in the form of relations to create a KG based on hand-crafted patterns. Recent work [61] applies neural relation extraction methods to directly generate relational facts from a given corpus and integrate them into an existing KG. During the integration process, these methods train a classifier to judge the correctness of the obtained relations according to the given KG. However, although the current Open IE methods extract relational facts (triples) directly from text sources, few of them address how to automatically merge the obtained facts to create a uniform and high-quality KG.

6 Conclusion and Future Work

This paper has proposed a novel Collaborative Knowledge Graph Fusion framework to integrate the joint event extraction and the knowledge graph fusion tasks together. The implemented prototype system with the proposed framework can both extract the entity and trigger text mentions and enrich the extracted mentions to a knowledge graph in the form of the knowledge graph triple (entity-relation-entity). To this end, we propose a benchmark-based supervision mechanism to guide the event extraction process of our system with a given knowledge graph. Our system also merges the extracted triples to the target knowledge graph by referring to the proposed Translated Relation Alignment Score. We test our prototype system on several real-world corpora and knowledge graphs. The experimental results show that our method improves the performances of both the event extraction and knowledge graph fusion processes after the alternative training. Moreover, the aligned and

translated relations from our system also show good interpretability. Our future work will aim to align the triples directly with their semantic meanings to further improve the performance of our model.

REFERENCES

[1] X. Wang, X. He, Y. Cao, M. Liu, and T. Chua, "KGAT: knowledge graph attention network for recommendation," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, A. Teredesai, V. Kumar, Y. Li, R. Rosales, E. Terzi, and G. Karypis, Eds. ACM, 2019, pp. 950–958.

[2] K. Annervaz, S. B. R. Chowdhury, and A. Dukkipati, "Learning beyond datasets: Knowledge graph augmented neural networks for natural language processing," in *Proceedings of NAACL-HLT*, 2018, pp. 313–322.

[3] A. Talmor and J. Berant, "The web as a knowledge-base for answering complex questions," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, M. A. Walker, H. Ji, and A. Stent, Eds. Association for Computational Linguistics, 2018, pp. 641–651.

[4] S. Hu, L. Zou, J. X. Yu, H. Wang, and D. Zhao, "Answering natural language questions by subgraph matching over knowledge graphs," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 5, pp. 824–837, 2018.

[5] W. Shen, Y. Yin, Y. Yang, J. Han, J. Wang, and X. Yuan, "Toward tweet entity linking with heterogeneous information networks," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2021.

[6] D. Vrandecic and M. Krötzsch, "Wikidata: a free collaborative knowledgebase," *Commun. ACM*, vol. 57, no. 10, pp. 78–85, 2014.

[7] G. A. Miller, "Wordnet: A lexical database for english," *Commun. ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[8] K. D. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: a collaboratively created graph database for structuring human knowledge," in *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008*, J. T. Wang, Ed. ACM, 2008, pp. 1247–1250.

[9] M. Färber, F. Bartscherer, C. Menne, and A. Rettinger, "Linked data quality of dbpedia, freebase, opencyc, wikidata, and YAGO," *Semantic Web*, vol. 9, no. 1, pp. 77–129, 2018.

[10] J. Li, A. Sun, J. Han, and C. Li, "A survey on deep learning for named entity recognition," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2020.

[11] D. Liu, T. Bai, J. Lian, X. Zhao, G. Sun, J. Wen, and X. Xie, "News graph: An enhanced knowledge graph for news recommendation," in *KaRS@CIKM 2019, Beijing, China, November 7, 2019*, ser. CEUR Workshop Proceedings, vol. 2601. CEUR-WS.org, 2019, pp. 1–7.

[12] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," in *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, K. Knight, A. Nenkova, and O. Rambow, Eds. The Association for Computational Linguistics, 2016, pp. 260–270.

[13] Y. Lin, S. Shen, Z. Liu, H. Luan, and M. Sun, "Neural relation extraction with selective attention over instances," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics, 2016.

[14] M. Koutraki, N. Preda, and D. Vodislav, "Online relation alignment for linked datasets," in *The Semantic Web - 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 - June 1, 2017, Proceedings, Part I*, ser. Lecture Notes in Computer Science, E. Blomqvist, D. Maynard, A. Gangemi, R. Hoekstra, P. Hitzler, and O. Hartig, Eds., vol. 10249, 2017, pp. 152–168.

[15] X. Zhao, Y. Jia, A. Li, R. Jiang, and Y. Song, "Multi-source knowledge fusion: a survey," *World Wide Web*, vol. 23, no. 4, pp. 2567–2592, 2020.

[16] Y. Wang, Z. Xu, L. Bai, Y. Wan, L. Cui, Q. Zhao, E. R. Hancock, and P. S. Yu, "Cross-supervised joint-event-extraction with heterogeneous information networks," 2020.

[17] P. Huang, X. Zhao, R. Takanobu, Z. Tan, and W. Xiao, "Joint event extraction with hierarchical policy network," in *Proceedings of the 28th International Conference on Computational Linguistics*. Barcelona, Spain (Online): International Committee on Computational Linguistics, Dec. 2020, pp. 2653–2664.

[18] H. L. Nguyen, D. Vu, and J. J. Jung, "Knowledge graph fusion for smart systems: A survey," *Inf. Fusion*, vol. 61, pp. 56–70, 2020.

[19] X. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun, and W. Zhang, "Knowledge vault: a web-scale approach to probabilistic knowledge fusion," in *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, S. A. Macskassy, C. Perlich, J. Leskovec, W. Wang, and R. Ghani, Eds., 2014, pp. 601–610.

[20] X. L. Dong and D. Srivastava, "Knowledge curation and knowledge fusion: Challenges, models and applications," ser. SIGMOD '15. New York, NY, USA: Association for Computing Machinery, 2015, p. 2063–2066.

[21] J. Bleiholder and F. Naumann, "Data fusion," *ACM Comput. Surv.*, vol. 41, no. 1, jan 2009.

[22] X. L. Dong, E. Gabrilovich, G. Heitz, W. Horn, K. Murphy, S. Sun, and W. Zhang, "From data fusion to knowledge fusion," *Proc. VLDB Endow.*, vol. 7, no. 10, p. 881–892, jun 2014.

[23] R. Sourty, J. G. Moreno, F.-P. Servant, and L. Tamine-Lechani, "Knowledge base embedding by cooperative knowledge distillation," in *Proceedings of the 28th International Conference on Computational Linguistics*. Barcelona, Spain (Online): International Committee on Computational Linguistics, Dec. 2020, pp. 5579–5590.

[24] P. Cui, X. Wang, J. Pei, and W. Zhu, "A survey on network embedding," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 5, pp. 833–852, 2019.

[25] R. Socher, D. Chen, C. D. Manning, and A. Y. Ng, "Reasoning with neural tensor networks for knowledge base completion," in *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, C. J. C. Burges, L. Bottou, Z. Ghahramani, and K. Q. Weinberger, Eds., 2013, pp. 926–934.

[26] Q. Wang, B. Wang, and L. Guo, "Knowledge base completion using embeddings and rules," in *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, Q. Yang and M. J. Wooldridge, Eds. AAAI Press, 2015, pp. 1859–1866.

[27] S. Guan, X. Jin, Y. Wang, and X. Cheng, "Shared embedding based neural networks for knowledge graph completion," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, A. Cuzzocrea, J. Allan, N. W. Paton, D. Srivastava, R. Agrawal, A. Z. Broder, M. J. Zaki, K. S. Candan, A. Labrinidis, A. Schuster, and H. Wang, Eds. ACM, 2018, pp. 247–256.

[28] A. Bordes, N. Usunier, A. García-Durán, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, C. J. C. Burges, L. Bottou, Z. Ghahramani, and K. Q. Weinberger, Eds., 2013, pp. 2787–2795.

[29] T. Dettmers, M. Pasquale, S. Pontus, and S. Riedel, "Convolutional 2d knowledge graph embeddings," in *Proceedings of the 32th AAAI Conference on Artificial Intelligence*, February 2018, pp. 1811–1818.

[30] D. Q. Nguyen, T. D. Nguyen, D. Q. Nguyen, and D. Phung, "A novel embedding model for knowledge base completion based on convolutional neural network," in *Proceedings of the 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2018, pp. 327–333.

[31] "Ace 2005, linguistic data consortium," http://projects.ldc.upenn.edu/ace.

[32] B. D. Trisedya, J. Qi, and R. Zhang, "Entity alignment between knowledge graphs using attribute embeddings," in *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press, 2019, pp. 297–304.

[33] D. Zeng, K. Liu, Y. Chen, and J. Zhao, "Distant supervision for relation extraction via piecewise convolutional neural networks," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, L. Màrquez, C. Callison-Burch, J. Su, D. Pighin, and Y. Marton, Eds. The Association for Computational Linguistics, 2015, pp. 1753–1762.

[34] S. Ji, S. Pan, E. Cambria, P. Marttinen, and P. S. Yu, "A survey on knowledge graphs: Representation, acquisition and applications," *CoRR*, vol. abs/2002.00388, 2020. [Online]. Available: https://arxiv.org/abs/2002.00388

[35] B. Yang and T. M. Mitchell, "Joint extraction of events and entities within a document context," in *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, K. Knight, A. Nenkova, and O. Rambow, Eds. The Association for Computational Linguistics, 2016, pp. 289–299.

[36] R. Qiu, Z. Huang, H. Yin, and Z. Wang, "Contrastive learning for representation degeneration problem in sequential recommendation," in *WSDM '22: The Fifteenth ACM International Conference on Web Search and Data Mining, Virtual Event / Tempe, AZ, USA, February 21 - 25, 2022*, K. S. Candan, H. Liu, L. Akoglu, X. L. Dong, and J. Tang, Eds. ACM, 2022, pp. 813–823.

[37] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, J. Burstein, C. Doran, and T. Solorio, Eds. Association for Computational Linguistics, 2019, pp. 4171–4186.

[38] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "Bpr: Bayesian personalized ranking from implicit feedback," 2012. [Online]. Available: https://arxiv.org/abs/1205.2618

[39] S. Arora, Y. Liang, and T. Ma, "A simple but tough-to-beat baseline for sentence embeddings," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

[40] B. Hu, Z. Lu, H. Li, and Q. Chen, "Convolutional neural network architectures for matching natural language sentences," in *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., 2014, pp. 2042–2050.

[41] T. Dettmers, P. Minervini, P. Stenetorp, and S. Riedel, "Convolutional 2d knowledge graph embeddings," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[42] Z. Zhang, Z. Li, H. Liu, and N. N. Xiong, "Multi-scale dynamic convolutional network for knowledge graph embedding," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 5, pp. 2335–2347, 2022.

[43] T. N. Kipf, E. van der Pol, and M. Welling, "Contrastive learning of structured world models," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[44] C. Gardent, A. Shimorina, S. Narayan, and L. Perez-Beltrachini, "Creating training corpora for NLG micro-planners," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, R. Barzilay and M. Kan, Eds. Association for Computational Linguistics, 2017, pp. 179–188.

[45] "Conll 2002, spanish efe news agency," https://www.clips.uantwerpen.be/conll2002/ner/.

[46] E. Sandhaus, "The new york times annotated corpus, publish linguistic data consortium, philadelphia 2008," in *publish Linguistic Data Consortium*, 2008.

[47] Q. Li, H. Ji, and L. Huang, "Joint event extraction via structured prediction with global features," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers*. The Association for Computer Linguistics, 2013, pp. 73–82.

[48] N. Limsopatham and N. Collier, "Bidirectional LSTM for named entity recognition in twitter messages," in *Proceedings of the 2nd Workshop on Noisy User-generated Text, NUT@COLING 2016, Osaka, Japan, December 11, 2016*, B. Han, A. Ritter, L. Derczynski, W. Xu, and T. Baldwin, Eds. The COLING 2016 Organizing Committee, 2016, pp. 145–152.

[49] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, A. Moschitti, B. Pang, and W. Daelemans, Eds. ACL, 2014, pp. 1532–1543.

[50] T. M. Nguyen and T. H. Nguyen, "One for all: Neural joint modeling of entities and events," in *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press, 2019, pp. 6851–6858.

[51] S. Zhang, P. Ng, Z. Wang, and B. Xiang, "Reknow: Enhanced knowledge for joint entity and relation extraction," *CoRR*, vol. abs/2206.05123, 2022.

[52] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.

[53] M. D. Zeiler, "ADADELTA: an adaptive learning rate method," *CoRR*, vol. abs/1212.5701, 2012.

[54] Y. Lin, H. Ji, F. Huang, and L. Wu, "A joint neural model for information extraction with global features," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, D. Jurafsky, J. Chai, N. Schluter, and J. R. Tetreault, Eds. Association for Computational Linguistics, 2020, pp. 7999–8009.

[55] Y. Lu, H. Lin, J. Xu, X. Han, J. Tang, A. Li, L. Sun, M. Liao, and S. Chen, "Text2event: Controllable sequence-to-structure generation for end-to-end event extraction," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds. Association for Computational Linguistics, 2021, pp. 2795–2806.

[56] Q. Wang, Z. Mao, B. Wang, and L. Guo, "Knowledge graph embedding: A survey of approaches and applications," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 12, pp. 2724–2743, 2017.

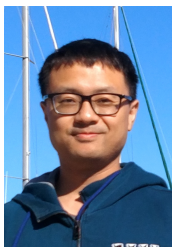
[57] Z. Li, H. Liu, Z. Zhang, T. Liu, and N. N. Xiong, "Learning knowledge graph embedding with heterogeneous relation attention networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 8, pp. 3961–3973, 2022.

[58] C. Shi, Y. Lu, L. Hu, Z. Liu, and H. Ma, "Rhine: relation structure-aware heterogeneous information network embedding," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 1, pp. 433–447, 2020.

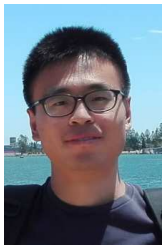
[59] G. Angeli, M. J. J. Premkumar, and C. D. Manning, "Leveraging linguistic structure for open domain information extraction," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*. The Association for Computer Linguistics, 2015, pp. 344–354.

[60] Mausam, "Open information extraction systems and downstream applications," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, S. Kambhampati, Ed. IJCAI/AAAI Press, 2016, pp. 4074–4077.

[61] B. D. Trisedya, G. Weikum, J. Qi, and R. Zhang, "Neural relation extraction for knowledge base enrichment," in *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, A. Korhonen, D. R. Traum, and L. Màrquez, Eds. Association for Computational Linguistics, 2019, pp. 229–240.



Yue Wang received the Ph.D. degree from Sichuan University, Sichuan, China. He was a postdoctor at Peking University, Beijing, China. He is now an Associate Professor at the Central University of Finance and Economics, Beijing, China. He has published more than 30 journal and conference papers, including TKDE, WWWJ, Science China: Information Science, IJCAI, ICDM, IEEE BigData, etc. His current research interests include data mining and machine learning.



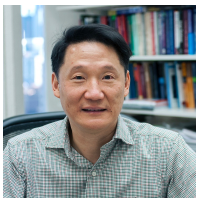
Yao Wan received his Ph.D. degree from the College of Computer Science, Zhejiang University, Hangzhou, China, in 2019. He is currently a lecturer at the College of Computer Science and Technology, Huazhong University of Science and Technology. He has been a visiting student of the University of Technology Sydney and the University of Illinois at Chicago in 2016 and 2018, respectively. His research interests lie in the synergy between artificial intelligence and software engineering, especially natural language processing, programming languages, software engineering, and machine learning.

Lu Bai received the BSc and MSc degrees from the Macau University of Science and Technology, Macau SAR, China, and the Ph.D. degree from the University of York, U.K. He was a recipient of the National Award for Outstanding Self-Financed Chinese Students Study Aboard by the China Scholarship Council, in 2015, and the best paper awards of the International Conferences ICIAP 2015 (Eduardo Caianello Best Student Paper Award) and ICPR 2018. He is now supported by the National Excellent Young Scientist Fund of NSFC. He was selected as one of the 2022 Baidu Global Top Chinese Young Scholars in Artificial Intelligence. He is now a Full Professor at the School of Artificial Intelligence, Beijing Normal University, Beijing, China, as well as the Central University of Finance and Economics, Beijing, China. He has published nearly 100 journal and conference papers, including TPAMI, TKDE, TNNLS, TCYB, TITS, PR, ICML, IJCAI, ICDE, ECML-PKDD, ICDM, CIKM, etc. His current research interests include pattern recognition, machine learning, and financial data analysis. He is currently a member of the editorial board of the journal Pattern Recognition.

Lixin Cui received the BSc and MSc degrees from Tianjin University, Tianjin, China, and the Ph.D. degree from the University of Hong Kong, HKSAR, China. She is now an associate professor at the Central University of Finance and Economics, Beijing, China. She was the recipient of the outstanding paper awards of the International Conference IEEE IEEM2019, the best student paper awards of the International Conferences APIEMS 2011 and WCE 2011, and the best scientific paper award of the International Conference ICPR 2018. She has published more than 40 journal and conference papers, including TPAMI, TFS, TNNLS, TKDE, TCYB, PR, IJPR, JIM, WWWJ, ICML, IJCAI, ECML-PKDD, etc. Her current research interests include machine learning, optimization algorithms, deep learning, and their applications in Fintech problems. She is currently a member of the editorial board of the journal Pattern Recognition.

Zhuo Xu received the B.Sc. degrees from Central University of Finance and Economics. He is now a graduate explorer at the Central University of Finance and Economics, Beijing, China.

Ming Li is currently a "Shuang Long Scholar" Distinguished Professor at the Key Laboratory of Intelligent Education Technology and Application of Zhejiang Province, Zhejiang Normal University, China. He received his Ph.D. degree from the Department of Computer Science and IT at La Trobe University, Australia. He completed two Postdoctoral Fellowship positions with the Department of Mathematics and Statistics, La Trobe University, Australia, and the Department of Information Technology in Education, South China Normal University, China, respectively. He has published in top-tier journals and conferences, including Artificial Intelligence, IEEE TCYB, IEEE TII, ACM TMOS, NeurIPS, ICML. He, as a leading guest editor, organized a special issue "*Deep Neural Networks for Graphs: Theory, Models, Algorithms and Applications*" in IEEE TNNLS. He is a PC member at ICML, AAAI, NeurIPS, ICLR, AJCAI, KDD, and an Associated Editor of Neural Networks.



Philip S. Yu received the B.S. degree in electrical engineering from National Taiwan University, the M.S. and Ph.D. degrees in EE from Stanford University, and the MBA degree from New York University. He is currently a Distinguished Professor of computer science at the University of Illinois at Chicago (UIC) and holds the Wexler Chair in information technology. He has published more than 970 papers in refereed journals and conferences. He holds or has applied for over 300 US patents. He was a member of the Steering Committee of the IEEE Data Engineering and the IEEE Conference on Data Mining. He is a Fellow of the ACM and the IEEE. He is on the Steering Committee of the ACM Conference on Information and Knowledge Management. He received the ACM SIGKDD 2016 Innovation Award for his influential research and scientific contributions on mining, fusion, and anonymization of big data, the IEEE Computer Society's 2013 Technical Achievement Award for "pioneering and fundamentally innovative contributions to the scalable indexing, querying, searching, mining, and anonymization of big data", and the Research Contributions Award from ICDM 2003, for his pioneering contributions to the field of data mining. He also received the ICDM 2013 10-year Highest-Impact Paper Award and the EDBT Test of Time Award (2014). He has received several IBM honors, including two IBM Outstanding Innovation Awards, an Outstanding Technical Achievement Award, two Research Division Awards, and the 94th plateau of Invention Achievement Awards. He was the Editor-in-Chief of the IEEE Transactions on Knowledge and Data Engineering (2001-2004).

Edwin R. Hancock (Fellow, IEEE) received the B.Sc. degree in physics, the Ph.D. degree in high-energy physics, and the D.Sc. degree from Durham University, Durham, U.K., in 1977, 1981, and 2008, respectively, and the Doctorate Honoris Causa from the University of Alicante, Alicante, Spain, in 2015. He is currently an Emeritus Professor with the Department of Computer Science, University of York, York, U.K., an Adjunct Professor with Beihang University, Beijing, China, a Distinguished Visiting Professor with Xiamen University, Xiamen, China and a Shanghai Science and Technology Commission Overseas Visiting Fellow at Shanghai University. His main research interests are in pattern recognition, machine learning, and computer vision, in which he has made sustained contributions to the use of graph-based methods and physics-based vision. Dr. Hancock was the 2016 Distinguished Fellow of the British Machine Vision Association (BMVA). He was elected as a fellow of the Royal Academy of Engineering (the U.K.'s National Academy of Engineering) in 2021. He is also a fellow of the IEEE and the International Association for Pattern Recognition (IAPR). He has also been a member of the editorial boards of the following journals: IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, Pattern Recognition, Computer Vision and Image Understanding, Image and Vision Computing, and the International Journal of Complex Networks. He was a recipient of the Pattern Recognition Medal in 1992, the IAPR Piero Zamperoni Award in 2006, the Royal Society Wolfson Research Merit Award in 2008, and the IAPR Pierre Devijver Award in 2018. He was the Founding Editor-in-Chief of IET Computer Vision from 2006 to 2012 and is currently the Editor-in-Chief of the journal Pattern Recognition. He was the Vice President of the IAPR from 2016 to 2018. He is an IEEE Computer Society Distinguished Visitor for the period 2021–2023.