ORIGINAL ARTICLE

Document-level relation extraction based on sememe knowledge-enhanced abstract meaning representation and reasoning

Qihui Zhao[1] · Tianhan Gao[1] · Nan Guo[2]

Abstract

Document-level relation extraction is a challenging task in information extraction, as it involves identifying semantic relations between entities that are dispersed throughout a document. Existing graph-based approaches often rely on simplistic methods to construct text graphs, which do not provide enough lexical and semantic information to accurately predict the relations between entity pairs. In this paper, we introduce a document-level relation extraction method called **SKAMRR** (**S**ememe **K**nowledge-enhanced **A**bstract **M**eaning **R**epresentation and **R**easoning). First, we generate document-level abstract meaning representation graphs using rules and acquire entity nodes' features through sufficient information propagation. Next, we construct inference graphs for entity pairs and utilize graph neural networks to obtain their representations for relation classification. Additionally, we propose the global adaptive loss to address the issue of long-tailed data. We conduct extensive experiments on four datasets DocRE, CDR, GDA, and HacRED. Our model achieves competitive results and its performance outperforms previous state-of-the-art methods on four datasets.

Keywords Document-level relation extraction · Graph neural networks · Sememe computation · Abstract meaning representation · Long-tailed task

Introduction

Information extraction is a fundamental task in natural language processing (NLP) and relation extraction is an important sub-task in information extraction [1, 2]. An early application scenario is to extract the relation between two entities in a single sentence, and this kind of work has been successful [3–5]. In recent years, the more challenging document-level relation extraction (DRE) has attracted the attention of scholars, and more work has started to focus on this task [6]. DRE plays a crucial role in knowledge acquisition of unstructured documents, facilitating many NLP

✉ Nan Guo
guonan@mail.neu.edu.cn

Qihui Zhao
1910459@stu.neu.edu.cn

Tianhan Gao
gaoth@mail.neu.edu.cn

[1] Software College, Northeastern University, Chuangxin, Shenyang 110819, Liaoning, China

[2] School of Computer Science and Engineering, Northeastern University, Chuangxin, Shenyang 110819, Liaoning, China

down-stream tasks such as knowledge graphs, recommendation systems, and semantic search (Fig. 1).

Many previous document-level relation extraction methods use dependencies and basic rules to generate document graphs and then use graph neural networks to make the inference, such as EoG [7], GAIN [8], DRN [9], and LSR [10]. However, these approaches have drawbacks. For example, the graph structure they exploit suffers from inadequate extraction of semantic information. Some works try to solve the task directly using language models (transformer, etc.) [11, 12]. These works usually take an equal view of the mention of different entities and suffer from the problem of over simplicity in the way they fuse features of the same entities. A few works focus on the distinction between positive and negative samples but not on the problem of category imbalance. Finally, only very little work currently considers the problem of lexical confusion which is a crucial issue in semantic-based NLP tasks.

The differences between document-level relation extraction and sentence-level relation extraction are mainly the following three points. (1) The first point is that entities are richer and distributed in long texts, and the same entities may appear in different sentences. We need more efficient ways to

Fig. 1 An example of DocRED

model the entities in a document that need to determine relationships. (2) The second point is that the long text increases the difficulty of reasoning about the relations between entities. For example, some relations must be extracted across five or more sentences for reasoning. Therefore, we need to use more precise reasoning to infer the corresponding relation labels. (3) The third point is that the data distribution is very uneven. For example, each document in DocRED [6] is annotated with named entity mentions, co-reference information, intra- and inter-sentence relations, and supporting evidence. There are about 10% of the relation types occupy nearly 60% of the total sample size. How to alleviate the uneven sample distribution is a key point for improving performance on this task.

To solve the above three problems, we propose a novel document-level relation extraction method, dubbed SKAMRR (sememe knowledge-enhanced abstract meaning representation and reasoning), as illustrated in Fig. 2. The model builds on PLM, constructs document-level AMR graphs, and models effective semantic associations. The model also constructs a document-level entity graph to reason correctly about the entity-pair belonging relations. Furthermore, the approach in this paper devises new loss functions to mitigate the problem of uneven distribution of data in the dataset. Intuitively, the document-level AMR graph is a core information extraction graph in which document nodes, sentence nodes, AMR nodes, and entity mention nodes are connected by different types of edges to simulate clustering information. Relation inference is an operation built on the entity mention graph (where entity mentions nodes are obtained from the document-level AMR graph) and is a semantically active graph designed to model the information about the relations that exist between different entity pairs.

First, we fuse the sememe information into word representation including entity mentions which can alleviate the problem of lexical confusion and generate sentence-level abstract meaning representation (AMR) graphs based on the sentences in a document and then generate document-level AMR graphs based on the rules. Document-level AMR graphs are rooted, annotated, directed, and acyclic graphs that represent high-level semantic relationships between abstract concepts of unstructured concrete natural text. AMR is a high-level semantic abstraction. In particular, different sentences that are semantically similar may share the same AMR parse output, which may also automatically filter and exclude some information that is unnecessary to the model to some degree.

Second, we build entity-pair graphs and use graph neural networks (GNN) to obtain vector representations of the nodes in the graph. Specifically, we obtain the representation of entity mention nodes from the document-level AMR graph and construct the entity pair graph, which is the core inference graph of this method. As part of the inference graph construction process, in this paper, the same entity nodes in pairs of entity nodes are connected to realize the document-level inference graph (multi-hop inference). Then, SKAMRR obtains entity representations enhanced by core inference with contextual feature information through graph isomorphism network [13], which facilitates capturing long-range relation information.

Finally, we design a global adaptive loss function to solve the problem of long-tail data. In addition, the problem of uneven distribution of data in the document-level relation extraction dataset is particularly apparent in DocRED, the dataset often used for this task.
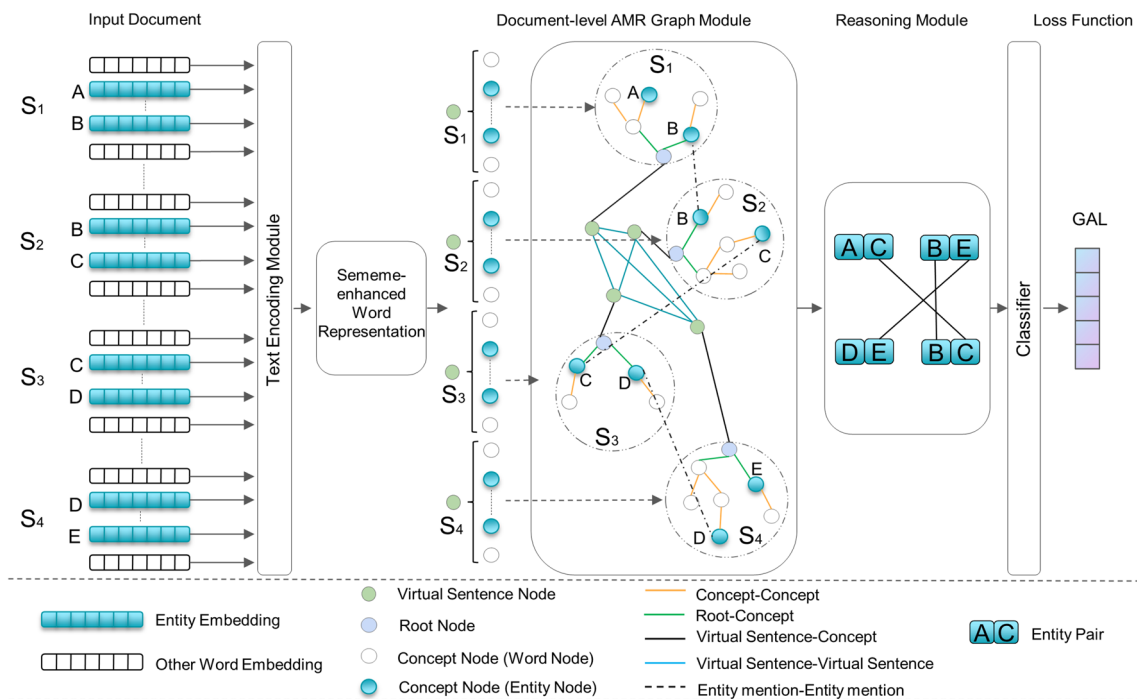
Fig. 2 The overall architecture of SKAMRR. First, the input document is subsequently encoded through BERT. Then, the sememe knowledge-enhanced AMR graph generates the head and tail entity representation. Next, we construct a entity-pair graph and use GIN to model the graph interaction. Finally, the classifier predicts relations of all the entity pairs and calculate model loss by GAL we proposed

The main contributions of this paper are summarized as follows:

- We construct a document-level AMR heterogeneous graph. This graph structure can well model the abstract semantic information at the document level. Because of its own advantages, it can automatically assist the model in performing the filtering of repetitive information.
- We introduce sememe knowledge in the document-level graph to improve the lexical representation of words at a finer level by merging different lexical senses and sememe information.
- We design a novel loss function named global adaptive loss (GAL). This function can mitigate the impact of long-tail effect on the model performance in the dataset and can improve the generalization ability of the model.
- SKAMRR outperforms the baseline models on four document-level relation extraction datasets (DocRED, CDR, GDA, and HacRed). Our experimental results demonstrate the efficacy of our method achieving competitive performance.

Organization. The rest of this paper is organized as follows. In the section "Related Work", we introduce the related work including document-level relation extraction, graph neural networks and abstract meaning representation. In the section "Our Method", we describe our proposed method in detail. The section "Experiments" gives the experimental setup and results. In the section "Conclusion", we conclude the entire paper.

Related work

Document-level relation extraction

Document-level relation extraction can generally be classified into two categories: document graph-based and sequence-based approaches. The graph-based approaches mainly use words or entities as graph nodes, construct the document graph by learning the latent graph structure of the document, and continue to infer using graph neural networks. Reference [14] first proposes the use of constructing document graphs to solve relation extraction across sentences. Reference [7] constructs heterogeneous graphs with three kinds of nodes and five kinds of edges. Reference [10] uses the matrix tree principle for heterogeneous networks to construct the same expression using the interaction of attention and iteratively updates the matrix by inducing structure. Reference [15] performs the DRE by learning a pronoun–mention graph representation, from which the derived graph can model the relation among pronouns and mentions to infer

the relations. References [8, 9, 16–21] all predict relations by constructing document graphs and devising a way to reason based on graph representations. There is another class of methods that mainly take a sequence-based model [19, 22]. As the transformer model has been used in the NLP field in recent years, more relation extraction methods have been applied to this model. Since the sequence-based transformer can model long-range sequences, such methods do not introduce graph structure. Reference [11] incorporates structural dependencies into the encoder network and can perform both context reasoning and structure reasoning. Reference [23] introduces a localized context pooling technique to solve the problem of using the same entity embedding for all entity pairs and proposes adaptive threshold Loss for long-tail data. Reference [24] proposes an entity knowledge injection framework to enhance DRE task by introducing co-reference distillation and representation reconciliation. Reference [25] proposes densely connected criss-cross attention network, which can collect contextual information in horizontal and vertical directions on the entity-pair matrix to enhance the corresponding entity-pair representation. Reference [26] builds upon co-reference resolution and gathers relevant signals via multi-instance learning. There has also been some recent work based on contrastive learning which focuses on issues of long tails and data noise in data sets. In this paper [27, 28], we use a graph structure-based approach, because graph structured data has a natural advantage for performing inference, both to accurately model documents and to capture semantic relationships between long-distance entities in more detail. To the best of our knowledge, we are the first to apply the AMR graph to the task of DRE.

Graph neural networks

The graph neural networks (GNN) have attracted increasing attention recently. While traditional neural networks are more suitable for data in Euclidean space, GNN can use neural networks in graph structures. There are many types of graph neural networks, including graph convolutional networks (GCN) [29], graph attention networks (GAT) [30], GIN [13], etc. GNN can be utilized in non-structural data where the graph structure is latent including the tasks of computer vision and natural language processing. In recent years, many works in NLP have applied for GNN techniques, such as text classification [31, 32], question answering [33, 34], text generation [35, 36], abnormal text detection [37, 38], etc. We employ three kinds of GNN to accomplish document-level relation extraction in our works. In this paper, we use graph neural networks to perform relational entity pair inference and fusion operations of sememe information.

HowNet and abstract meaning representation

HowNet is one of the most famous sememe knowledge bases, constructed in more than 20 years. A sememe is the minimum semantic unit in linguistics, and some linguists hold that the meanings of all words in a language can be represented by a limited set of sememes [39]. Hownet contains many Chinese and English words with word meanings and sememe information. The sememes of senses in HowNet are annotated with various relations and form hierarchical graph structures. In our work, we only consider all annotated sememes sets of each sense without considering their internal relations. HowNet proposes that annotated sememes can represent senses and words well in a real-world scenario. Sememes are helpful for many NLP tasks [40–42]. Open-HowNet API [43] is developed by THUNLP, which provides a convenient way to search information in HowNet, display sememe trees, calculate word similarity via sememes, etc. In our paper, we get words' senses and sememes by Open-HowNet API.

Abstract meaning representation (AMR) [44] is a graph-based semantic representation that captures the sentence's semantics of "who is doing what to whom". Each sentence is represented as an acyclic graph with labels on nodes (e.g., concepts) and edges (e.g., relations). Every node and edge of the graph are labeled according to the sense of the words in a sentence. An ID names each node in AMR. It contains the semantic concept, which can be a word (e.g., man) or a PropBank frameset (e.g., want-01) or a special keyword. The keywords have type (e.g., date-entity), quantities (e.g., distance-quantity), and logical conjunction (e.g., and). The edge between two nodes is annotated using more than 100 relations including frameset argument index (e.g., ":ARG0"), semantic relations (e.g., ":location"), etc. Several recent works in natural language processing use AMR [45–47]. In this paper, we have extended the sentence-level AMR to the document level for better adaptation to the task.

Our method

Research objective

The objective of this paper is to solve three RE problems at the document level: (1) the phenomenon of multiple meanings of a word plays an important role in the understanding of semantics, and the same problem exists in the dataset of document-level relation extraction. (2) There are many cross-sentence relations of entity pairs leading to long-distance dependencies. Accurate relation classification often requires strong graph modeling at the document level and a comprehensive reasoning approach. (3) Long-tail effect exists in the dataset leading to degraded model performance. We then

describe how to resolve these issues and present experimental analysis.

General framework of SKAMRR

In this section, we describe our model (SKAMRR) in detail. As shown in Fig. 2, the approach in this paper consists of four parts, (1) text encoding module: using a text encoder to obtain the initial word embedding. (2) Sememe knowledge-enhanced Abstract meaning representation (AMR) module: constructing document-level AMR graphs with Sememe-enhanced word representations and obtaining fully interacting word and entity nodes' representations. (3) Reasoning module: building entity-pair graph and performing relation reasoning. (4) Classification module: outputting relation by classification function and proposing global adaptive loss (GAL) to alleviate imbalance of the data.

Background and notation

We formulate the document-level RE as follows:

Document D: The document D is the raw text that contains multiple sentences. In addition, it makes use of a sequence of word tokens, $\{x_1, x_2, .., x_n\}$, to represent the input of word embedding.

Entity E and Mention m: The entity set E consists of the entities that appear in the document D. The mention represents the expression that does not have an explicit entity to refer to in a document, and each mention is defined to be a span of words. For each entity e_i, it is represented by a set of mentions: $ei = \{m_1, m_2, \ldots, m_n\}$ in the document D.

Formally, a document-level relation extraction task can be denoted as $T = \{X, Y\}$, where X is the instance set and Y is the relation labels set. For each instance, it consists of several tokens $\{x_1, x_2, \ldots, x_n\}$. The task aims to predict the relation labels between entities, namely $r_{h,t} = f(h_e^h, h_e^t)$, where h_e^h, h_e^t are the representations of head entity and tail entity in E, $r_{h,t}$ is a relation label.

Text encoding module

The pre-trained language model (PrLM) as the text encoder, such as BERT [48], is used in our work. BERT has achieved amazing results on several natural language processing tasks, demonstrating its powerful modeling ability for text data. Denoting a document D of length l as the input and $D = [x_t]^l$, where x_t means a word at position t. Following previous work, we add the special markers $< /S\text{-category} >$ and $< /E\text{-category} >$ before and after each entity mentions. Then, we can obtain the content embedding H

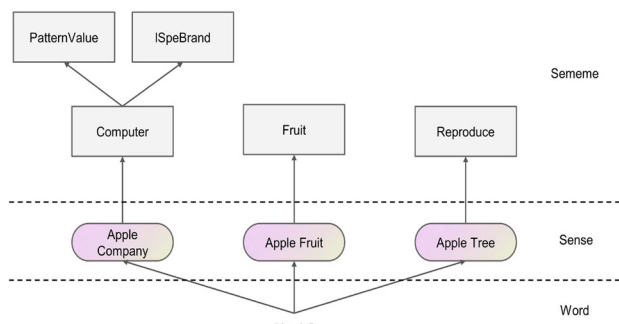$$H = \text{PrLM}([x_1, \ldots, x_n]) = [h_1, \ldots, h_n], \qquad (1)$$

Fig. 3 The word "apple" have three senses: apple company, apple fruit, and apple tree. "Apple Company" have three sememe information, including "PatternValue", "IspeBrand", and "Computer". "Apple Fruit" has "Fruit" and "Apple Tree" has "Reproduce"

where n is the length of the document after adding the special markers, and we concatenate the start and end markers of each entity mentions as its embedded representation. We use dynamic windows for long text ($n > 512$).

Sememe knowledge-enhanced abstract meaning representation module

Sememe-enhanced word representation

We introduce sememe knowledge to enhance the lexical representation of words at a more fine-grained level by fusing different lexical meanings and sememe information. As shown in Fig. 3, a word contains multiple senses, and a sense contains multiple sememes. The structure of the sememes of one sense is available to form a graph structure. First, we employ HowNet [39, 43] to get all words' sememes and senses information. Then, we construct the sememe graph for each word sense. We use the graph attention network (GAT) to pass and aggregate features on the sememe graph as the following equation:

$$\boldsymbol{h}_{\text{sem}_1}, \ldots, \boldsymbol{h}_{\text{sem}_M} = GAT(v_{\text{sem}_1}, \ldots, v_{\text{sem}_M}), \qquad (2)$$

where $\text{sem}_1, \ldots, \text{sem}_M$ denote all the sememes belonging to one sense sen, v_{sem} indicates the words embedding of sememe information, and $\boldsymbol{h}_{\text{sem}}$ represents the output of GAT.

We then calculate the representation of the sense by the averaging all sememe representations it have

$$\boldsymbol{h}_{\text{sen}} = \frac{\sum_{k=1}^{K} \boldsymbol{h}_{\text{sem}_k}}{K}. \qquad (3)$$

Afterward, all sense vectors are aggregated by global attention [49] to obtain new word representations

$$a_j = \frac{\exp(\tanh(w_s[\boldsymbol{h}_j; \boldsymbol{h}_{\text{sen}_j}]))}{\sum_{c=1}^{C} \exp(\tanh(w_s[\boldsymbol{h}_i; \boldsymbol{h}_{\text{sen}_c}]))} \qquad (4)$$

Fig. 4 An example of a document-level AMR graph, where S1, S2, and S3 are virtual sentence nodes; See-01, chase-01, call out-01 are root nodes; and the others are concept nodes

Sentence 1: I saw peter's cat, which was jumping at the window.

Sentence 2: The cat was chasing a mouse.

Sentence 3: Peter was calling out to his cat.
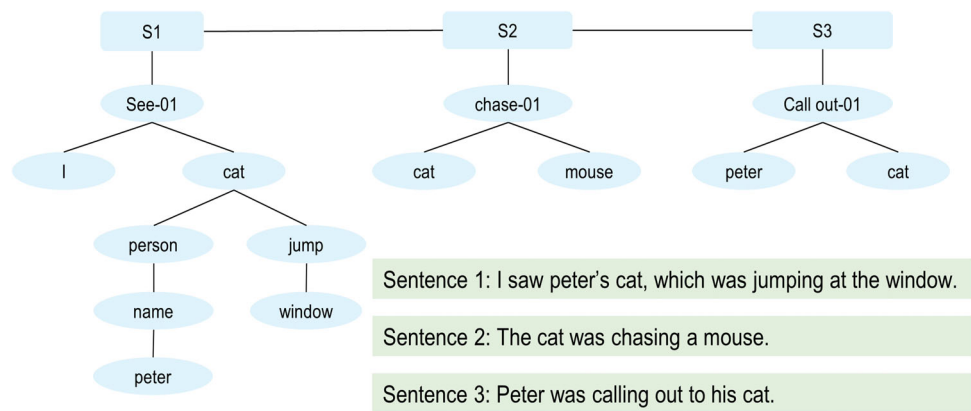
$$h_i^{\text{sem}} = \sum_{j=1}^{C} a_j h_{\text{sen}_j}, \tag{5}$$

where $\text{sen}_1, \ldots, \text{sen}_C$ denote the set of sense representations for the word i and C is the senses number of word i, w_s are trainable parameters, and $H^{\text{sem}} = h_1^{\text{sem}}, \ldots, h_n^{\text{sem}}$ are the sememe-enhanced representations and we use those as embedding vectors instead of the initial BERT outputs.

Building document-level AMR

AMR is an effective semantic formalism in nature language and can abstract the semantics of sentences to words that contain key information. Some recent works have demonstrated that AMR can assist in natural language processing tasks [47]. To obtain adequate and critical information for relation classification, we construct a document-level AMR graph $G^D = (V^D, E^D)$ for each document. The standard AMR graph is sentence-based, but the task is document-based and requires reasoning about the relationships between entities in different sentences across a document. First, we select all sentences in the document that contain entity mentions, and we use the AMR parsing model [50] to get the corresponding sentence-level AMR graph. The initial embedding of the node in the graph is the representation with the Sememe information fused in the previous step. For each sentence, we construct a virtual sentence node that is connected to the root node of the sentence-level AMR graph and all virtual sentence nodes are connected. A document-level AMR graph contains three types of nodes: root nodes, virtual sentence nodes, and concept (word) nodes; four types of edges: root node-virtual sentence nodes, virtual sentence node-virtual sentence node, concept node-concept node, and concept node-root node. The root nodes' embedding is average embedding of its sentence. The concept nodes' embedding is getting from sememe-enhanced word representation. The virtual sentence node embedding is calculated by the attention score of the last layer of BERT, which is calculated as follows:

$$a_i^{(h,t)} = \frac{A_i^h \cdot A_i^t}{\mathbf{1}^\top (A_i^h \cdot A_i^t)} \tag{6}$$

$$h_y^{\text{snode}} = H_y^{\text{sem}} a_i^{(h,t)}, \tag{7}$$

where A_i^t, A_i^h is the attention matrix for ith mention of head and tail entity tokens in one sentence. h_y^{snode} is the representation of yth virtual sentence node and $H^{\text{snode}} = h_1^{\text{snode}}, \ldots, h_Y^{\text{snode}}$. In addition, the mentions of the same entity are also connected with edges. For the few entity mentions that do not appear in the AMR graph, we construct additional entity nodes and connect them to the virtual sentence nodes of the sentences where they are in. Figure 4 is an example of the document-level AMR graph constructed in this paper.

Then, we use R-GCN [51] to perform feature extraction on each document-level AMR graphs

$$h_u^{(l+1)} = \text{ReLU}\left(\sum_{t \in \mathcal{T}} \sum_{v \in \mathcal{N}_u^t \cup \{u\}} \frac{1}{c_{u,t}} W_t^{(l)} h_v^{\text{sem}(l)}\right), \tag{8}$$

where $h_v^{\text{sem}} \in \{H^{\text{snode}}, H^{\text{sem}}\}$. $\mathcal{T}$ is an edge of different types and $W_t^l \in \mathbb{R}^{d*d}$ is a trainable parameter. $\mathcal{N}_u^t$ is the set of neighboring nodes of node u at an edge of type t. $c_{u,t} = |N_u^t|$ is a constant. Then, for the final representation of the node u, we use the following equation for calculation:

$$m_u = \text{RELU}(W_u \cdot [h_u^{(0)}; \ldots; h_u^{(N)}]) + \text{Max}(h_u^{(0)}, \ldots h_u^{(N)}), \tag{9}$$

where $W_u \in \mathbb{R}^{d*Nd}$ is the trainable parameter. m_u is the node u representation.

Reasoning module

Our approach fully interacts with the beneficial information in the document-level AMR in the previous module. In the inference module, we consider the multi-hop phenomenon (E.g., a document has four entities A, B, C, D, where (A, C) has relation a, (C, D) has relation b, and (D, B) has relation c. Then, it can be inferred that (A, B) has relation d.) of entities and entity mentions for document-level relation extraction. We employ connecting the first-order neighbors of entity mentions and generating inference graphs, and then use the GNN to obtain relation representations between multi-hop neighbors. We fuse multiple entity mentions of one entity, which is implemented using LogSumExp pooling [23] with the following equation:

$$\boldsymbol{h}_{e_i} = \log \sum_{j=1}^{N_{e_i}} \exp(\boldsymbol{m}_j), \tag{10}$$

where m_j denotes representation of jth mention of ith entity.

We construct the entity pairs graph for one document. In particular, one entities pair (head and tail entity) is regarded as a node. If one of the two entities contained in an entity-pair node is the same, then these two entity-pair nodes are connected. The formula for the feature representation of the entity-pair nodes is as follows:

$$\boldsymbol{h}_{(h,t)}^h = \tanh\left(W_h \boldsymbol{h}_e^h + b_h\right) \tag{11}$$

$$\boldsymbol{h}_{(h,t)}^t = \tanh\left(W_t \boldsymbol{h}_e^t + b_t\right) \tag{12}$$

$$\boldsymbol{h}_{(h,t)}^r = {\boldsymbol{h}_{(h,t)}^h}^T W_p \boldsymbol{h}_{(h,t)}^t, \tag{13}$$

where $W_h \in \mathbb{R}^{d*d}$, $W_t \in \mathbb{R}^{d*d}$, and $W_p^i \in \mathbb{R}^{d^2}$ are learnable parameters. $\boldsymbol{h}_{(h,t)}^r$ is the vector representation of entity-pair node.

Finally, we utilize a GNN to encode the entity pairs graphs to extract the relation information. Given a entity pairs graph g, representation after the graph encoder is as below

$$\boldsymbol{h}_{(h,t)_f}^1, \ldots, \boldsymbol{h}_{(h,t)_f}^S = G(\boldsymbol{h}_{(h,t)}^1, \ldots, \boldsymbol{h}_{(h,t)}^S), \tag{14}$$

where $G()$ is the graph encoder, and here, we use a state-of-the-art graph isomorphism network [13] for its strong representation ability. $\boldsymbol{h}_{(h,t)}^S$ denotes the initial node representations which are calculated above.

Classification module

We first concatenate the entity-pair representation and the two entity representations to generate the final representation for relation classification

$$\boldsymbol{r}_{(h,t)} = \left[\boldsymbol{h}_e^h; \boldsymbol{h}_e^t; \boldsymbol{h}_{(h,t)_f}\right], \tag{15}$$

where $\boldsymbol{h}_e^h$ and $\boldsymbol{h}_e^t$ are computed by Eq. (10). $\boldsymbol{h}_{(h,t)_f}$ is getting from Eq. (15).

Then, we adapt a linear layer for predicting relations

$$l_f^{(h,t)} = \mathbf{W}_f \boldsymbol{r}_{(h,t)} + b_f, \tag{16}$$

where $l_f \in \mathbb{R}^c$ denotes the output logits for all relations, $\mathbf{W}_f \in \mathbb{R}^{d \times c}$ is the weight matrix that maps the relation embedding to the each class, and c is the number of label categories.

Document-level relation extraction is essentially a multi-label classification problem, and [23] proposes adaptive thresholding loss (ATL) to solve the multi-label problem. ATL is designed with a special category TH as the adaptive threshold, with positive cases above TH and negative cases below or equal to TH. The original version of the loss function is formulated as follows:

$$P(r_i \mid e_h, e_t) = \frac{\exp\left(l_f^{(h,t)}\right)}{\sum_{r' \in \mathcal{P}_T \cup \{TH\}} \exp\left(l_f^{(h,t)'}\right)} \tag{17}$$

$$\mathcal{L}_1 = -\sum_{r \in \mathcal{P}_T} \log(P(r_i \mid e_h, e_t) \tag{18}$$

$$\mathcal{L}_2 = -\log\left(\frac{\exp\left(l_f^{(h,t)^{TH}}\right))}{\sum_{r' \in \mathcal{N}_T \cup \{TH\}} \exp\left(l_f^{(h,t)'}\right)}\right) \tag{19}$$

$$\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2, \tag{20}$$

where positive classes $\mathcal{P}_T \subseteq R$ are the relations that exist between the entities in T. If T does not express any relation, $\mathcal{P}_T$ is empty. Negative classes $\mathcal{N}_T \subseteq R$ are the relations that do not exist between the entities. If T does not express any relation, $\mathcal{N}_T = R$.

We use the idea of gradient harmonizing mechanism (GHM) [52] to balance the possibility of positive examples and propose gradient adaptive loss (GAL) to enhance the effect of ATL. Our loss function's design intuition is to keep the model from focusing more on hard-to-classify (outliers) and hard-to-classify samples. Gradient density is introduced to measure the number of samples appearing in a specific gradient range, so that the update of samples per gradient becomes more balanced

$$\mathcal{L}' = \alpha_1 \cdot \mathcal{L}_1 + \mathcal{L}_2 \tag{21}$$

$$\alpha_1 = \frac{N}{GD(g_r)}, \tag{22}$$

where $\mathcal{L}'$ is our loss named GAL. $GD(g) = \frac{1}{l_\varepsilon(g)}\sum_{k=1}^{N} \delta_\varepsilon(g_k, g)$, denotes the gradient density. $\delta_\varepsilon(g_k, g)$ denotes N

samples in each batch's slicing. The parameter α_1 enables to transform the adjustment of the gradient to the loss function. We can achieve the optimization for loss by adjusting the value of α_1.

Experiments

The goal of our experiments is to show that (1) our model can capture important sentence-level features as well as document-level features of relevant entity pairs and combine these features for inference to obtain document-level relation extraction results, and (2) our proposed loss function can mitigate the impact of the imbalanced sample distribution on the performance of the model. In this section, we first introduce four document-level relation extraction datasets. We then give some model parameters in this paper as well as the baseline model used for experimental comparison. We conclude by evaluating the model and using ablation experiments to illustrate the robustness and efficiency of the model architecture in this paper.

Dataset statistics

DocRED is a large-scale manually annotated document-level RE dataset constructed from Wikipedia and Wikidata with two features. (1) It contains 132,375 entities and 56,354 relationship facts annotated on 5,053 Wikipedia documents. (2) Since at least 40.7% of the relations in DocRED can only be extracted from multiple sentences, DocRED needs to read multiple sentences in a document to identify entities and reason about their relations. The dataset contains 3053, 1000, and 1000 instances as the training set, validation set, and test set, respectively. Reference [53] creates the Chemical-Disease Reactions dataset (**CDR**). It contains one kind of relation: Chemical-Induced Disease between chemical and disease entities. The dataset contains 500 documents for training, 500 for development, and 500 for testing. The Gene-Disease-Associations dataset (**GDA**) is created by [54]. It has one kind of relation which is "Gene-Induced-Disease" between gene and disease. We split the dataset in a normal method, 23,353 documents for training, 5839 for development, and 1000 for testing. **HacRED** is a large-scale dataset with reasonable data distribution which focus on the hard cases of relation extraction. We also select 6231 samples as the training set, 1500 as the validation set, and 1500 as the test set (Table 1).

Experiment settings and evaluation metrics

We use PyTorch [55] and DGL [56] frameworks to implement the model in this paper. For the DocRed dataset, we utilize BERT-large [48] and RoBERTa-large [57] as the ini-

Table 1 Dataset details

Dataset	DocRED	CDR	GAD	HacRED
Training set cases	3053	500	23,353	6231
Dev set cases	1000	500	5839	1500
Test set cases	1000	500	1000	1500
Relations	96	2	2	26
Avg. #Ents per Doc	19.5	7.6	2.4	10.8
Avg. #Ment. per Ent	1.4	2.7	3.3	1.2
Avg. #rel. per Doc	12.5	2.1	1.5	7.4

tial encoders for the documents, and Xu's model [50] as the AMR generator, respectively. For the CDR dataset, we use BioBERT-Base v1.1 as the encoder, and We employ the transformer-based AMR parser [58] that is pre-trained on the Biomedical AMR corpus. The model parameter optimizer we use is AdamW [59]. We set the initial learning rate for all encoder modules to 2e−5, other modules to 1e−4. We make the embedding dimension and the hidden dimension to 768. Our method's GNNs encoders have three layers and the hidden size of node embedding is 768. Our model is experimented with NVIDIA RTX 3090 GPU. Following previous work [6, 8], we take micro $F1$ and micro Ign $F1$ as the evaluation metrics for experimental performance. Ign $F1$ is the $F1$ metric after excluding the effect of the presence of the same entity relation pairs in the development/test set and the training set.

Compared methods

We compare multiple models, which can be classified into graph-based and non-graph-based approaches. We label Bert-base as **Bb**, Bert-large as **Bl**, and Roberta-large as **Rol**.

Graph-based methods:
LSR [10] is an end-to-end document-level relation extraction approach that treats the graph structure as a potential variable and corrects that graph at each iteration step.

GEDA [17] considers the attention between sentences and potential relation instances as a many-to-many relationship and therefore introduces a bi-attention mechanism, including the attention of sentence-to-relation and relation-to-sentence.

GCGCN-BERT [60] proposes a novel graph convolutional networks, which have two hierarchical blocks: context-aware attention guided graph convolution for partially connected graphs and multi-head attention guided graph convolution for fully connected graphs.

GLRE [16] models entity pairs by encoding document information into global and local representations as well as contextual relation representations.

HeterGSAN [19] manages to reconstruct path dependencies from graph representations to ensure that the proposed

DocRE model is more concerned with encoding pairs of entities with relations in training.

SIRE [21] represents intra- and inter-sentential relations differently and designs a straightforward form of logical reasoning that can cover more logical reasoning chains.

DRE [9] designs a discriminative inference network for estimating the relation probability distributions of different inference paths and then models the inference method for the relation between each entity pair in the document.

CGM2IR [61] proposes context-guided coreferential mentions integration in a weighted sum manner and inter-pair reasoning.

Non-graph-based methods:

BERT [62] solves the DRE task in phases that can improve performance, the first step is to predict whether two entities are related, and the second is to predict the specific relation.

HINBERT [63] proposes a hierarchical inference network (HIN) for document-level inference, which can aggregate inference information from entity level to sentence level and then document level.

CorefBERT [22] adds a mention reference prediction (MRP) pre-training task to achieve the purpose of fusing co-reference information in the pre-trained model.

SSAN [11] argues that structural dependencies should be incorporated within the encoding network and throughout the system, leading to the proposal of structured self-attention network, which can effectively model these dependencies within its construction blocks and in all network layers from the bottom up.

ATLOP [23] proposes localized context pooling structure and adaptive thresholding to solve the multi-label and multi-entity problem.

MRN [20] offers a mention-based reasoning network to distinguish the impacts of close and distant entity mentions in relation extraction and consider the interactions between local and global contexts.

DocuNet [12] analogize the DRE to the semantic segmentation task in computer vision, and use the U-shaped module to capture the global interdependencies between the triples on the image-style feature graph.

In CDR and GDA dataset, we compare our SKAMRR model with six baselines, including EoG [7], DHG [18], LSR, MRN, ATLOP, and CGM2IR.

Main results

Results on DocRED: We have conducted many experiments, and the results are presented in Tables 2, 3 and Figs. 5, 6. In DocRED, we can find that our model SKAMRR is better on both Dev and test with the baseline model. In the document graph-based approach, SKAMRR outperforms the best-performing model SIRE in both $F1$ and Ign $F1$ met-

Table 2 Results on the development and test set of DocRED

Models	Dev		Test	
	Ign $F1$	$F1$	Ign $F1$	$F1$
LSR-Bb [10]	52.43	59.00	56.97	59.05
GEDA-Bb [17]	54.52	56.16	53.71	55.74
GCGCN-BERT-Bb [60]	55.43	57.35	54.53	56.67
GLRE-Bb [16]	–	–	55.40	57.40
GAIN-Bb [8]	59.14	61.22	59.00	61.24
HeterGSAN-Bb [19]	58.13	60.18	57.12	59.45
SIRE-Bb [21]	59.82	61.60	60.18	62.05
DRE-Bb [9]	59.33	61.39	59.15	61.37
BERT-Bb [62]	–	54.16	–	53.20
HINBERT-Bb [63]	54.29	56.31	53.70	55.60
CorefBERT-Bb [22]	55.32	57.51	54.54	56.96
SSAN-Bb [11]	57.03	59.19	55.84	58.16
ATLOP-Bb [23]	59.22	61.09	59.31	61.30
MRN-Bb [20]	59.74	61.61	59.52	61.74
DocuNet-Bb [12]	59.86	61.83	59.93	61.86
CGM2IR-Bb [61]	60.02	62.01	60.24	62.06
SKAMRR-Bb	**62.08**	**64.05**	**62.27**	**64.16**
CorefBERT-Rol [22]	57.84	59.93	57.68	59.91
SSAN-Rol [11]	60.25	62.08	59.47	61.42
ATLOP-Rol [23]	61.32	63.18	61.39	63.40
GAIN-Bl [8]	60.87	63.09	60.31	62.76
CGM2IR-Bl [61]	62.03	63.95	61.96	63.89
SKAMRR-Rol	**64.06**	**66.97**	**64.96**	**67.14**

We separate Bb-based and Rol/Bl-based methods. The results of baselines are from their original papers

Values in bolded font are the metrics corresponding to the best performing models

Table 3 Experimental results on HacRED

Model	Precision	Recall	$F1$
LSR	70.12	68.94	69.45
GAIN	73.35	**79.98**	76.01
ATLOP	76.74	78.27	77.52
SKAMRR	**77.94**	78.89	**78.29**

All results are implemented by us and used RoBerta as the text encoder

Values in bolded font are the metrics corresponding to the best performing models

rics when using BERT-base as the document encoder, and our model SKAMRR outperforms the GAIN model when using RoBERTa-large as the document encoder. The best-performing models in document-level relation extraction are sequence-model-based approaches, represented by DocuNet and ATLOP. Our model SKAMRR outperforms both BERT-base and RoBERTa-large in comparison metrics $F1$ and Ign $F1$ when they are the document encoders, respectively. When BERT-base is the encoder, SKAMRR outperforms DocuNet
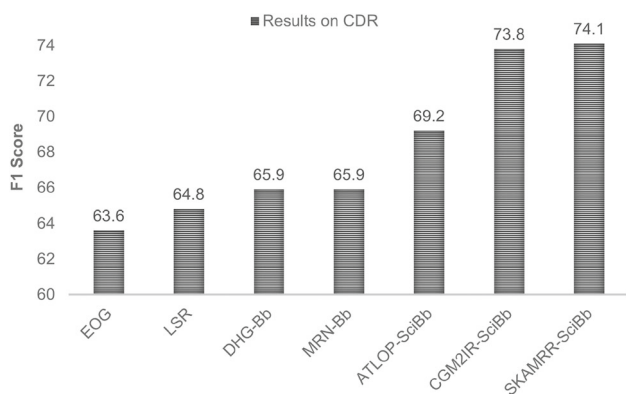
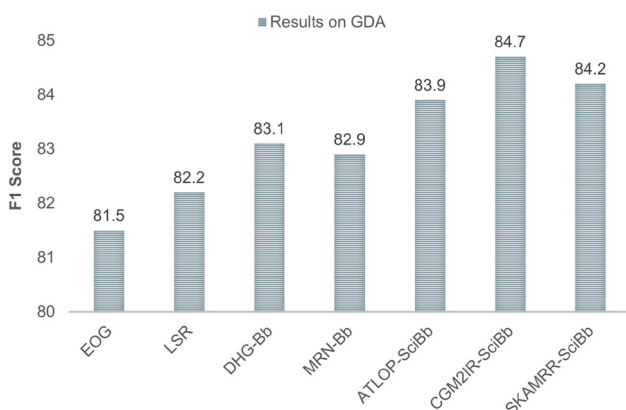**Fig. 5** Results on the test set of CDR. The results of baselines are from their original papers

Fig. 6 Results on the test set of GDA. The results of baselines are from their original papers

by 2.3% and 2.34% on $F1$ and Ign $F1$ in test, respectively. When RoBERTa-large is the encoder, SKAMRR outperforms ATLOP by 4.38% and 4.65% on $F1$ and Ign $F1$ in test, respectively. The gap between Ign $F1$ and $F1$ woes is also smaller when document encoders are used in this paper's model, showing that SKAMRR has good generalization and generality. Also, the performance when using RoBERTa-large as the encoder is better than using Bert-base, which shows the power of the pre-trained model, and the task will gain with the development of the pre-trained model at a later stage.

Results on CDR and GDA: The results on the CDR dataset by $F1$ score are shown in Fig. 5. It can be observed that among the methods, the graph-based methods (CGM2IR and ours) perform better in extracting the relation. These phenomena demonstrate that the graph structure can better preserve the interaction between different elements in the document, which can help the model to correctly classify the cross-sentence relation. Furthermore, our method achieve the best performance on all the metrics, which demonstrates the effectiveness of SKAMRR. In particular, it states that

not only can our method automatically learn multi-hop paths for inter-sentence relationships, but also identify the semantic path within the sentence for intra-sentence extraction. As can be observed from the experimental results presented in Fig. 6, our SKAMRR achieves 84.2 on the GDA, which is also better than nearly all of the methods. On the other hand, the SKAMRR metric is slightly. smaller than CGM2IR (-0.5%), which is primarily due to the presence of fewer inter-sentence relations in the GDA dataset (only 13% compared to 30% in the CDR dataset), which results in the under learning of the SKAMRR model. The method is also effective for document-level relation extraction in the biomedical domain.

Results on HacRED: The experimental results based on the HacRED dataset are shown in Table 3. In this paper, we have selected three baseline methods; ATLOP and GAIN, which represent the best performance of the graph-free and graph-based methods, respectively. We are decorrelating to the open source code supplied in the original paper for this experiment. Our proposed method is able to outperform the ATLOP baseline in all metrics, higher than 1.2% in accuracy, 0.62% in recall, and 0.77% in $F1$ value, respectively. Not only does the model perform well, but for all the methods compared, HacRED's performance is clearly superior to its performance on DocRED. Normally, because the HacRED dataset focuses on hard relations, while the DocRED dataset is more general, the model should be less effective. The reasons for this phenomenon are as follows: (1) the HacRED dataset has significantly more samples with annotations than DocRED, which also makes the model more fully trained and makes the model have better generalization. 2) There are only 26 relation categories in HacRED and the data are highly distributed, which significantly reduces the presence of fewer sample data, also making the model more fully trained.

Furthermore, we utilize four models, namely LSR, GAIN, ATLOP, and SKAMRR, to create the critical distance diagram. Based on Fig. 7, it is evident that the SKAMRR, which is introduced in this paper, outperforms the other models. The model in this paper achieves competitive results in all the four datasets. Experimental results show that the model of this paper can explore feature information well both within and across sentences, and accurately infer classes of relations between entities by the inference method devised in this paper.

Ablation study

We design the corresponding ablation experiments for the structure and contribution of our method. Our ablation experiments are divided into two parts. First, we perform the experiment for model's structure, involving the AMR mod-
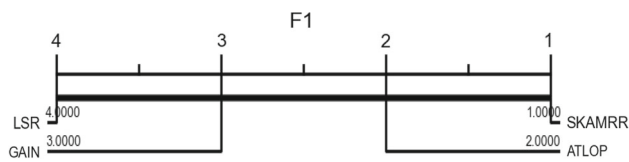
Fig. 7 Critical distance diagram for LSR, GAIN, ATLOP, and SKAMRR

Table 4 Ablation study of SKAMRR on the Dev set of DocRED, where "w/o" indicates without

Model	Ign $F1$	$F1$
SKAMRR-Bb	62.08	64.05
w/o AMR graph structure	60.71 ($-$ **1.37**)	62.65 ($-$ **1.40**)
w/o Sememe knowledge	60.97 ($-$ 1.11)	62.93 ($-$ 1.12)
w/o Global adaptive loss	60.94 ($-$ 1.14)	62.93 ($-$ 1.12)
w/o all	60.15 ($-$ **1.93**)	61.98 ($-$ **2.07**)

Bolded values indicate that each part of the proposed method highlighting this paper is effective, and if removed, the model performance decreases more

Table 5 Results for long-tail type relations which contains ten relations, where "w" indicates with

Model	Long-tail $F1$
SKAMRR-Bb	63.08
w CE	62.17 ($-$ 0.91)
w Adaptive thresholding loss	62.43 ($-$ 0.65)

Fig. 8 Results of different number of interactions on the test set of DocRED

in this paper can improve the model's performance on the class sample imbalanced dataset and mitigate the impact of the imbalanced sample distribution (Table 5).

Parameter sensitivity

Since we use GNN in sememe information fusion, AMR graph modeling, and entity-pair graph inference, we need to experimentally validate with respect to the key parameter in GNN (the number of interaction steps). The performance of our SKAMRR method is influenced by the number of interaction steps, so we can choose multiple numbers to verify our approach. This part compares different numbers of interaction steps to analyze which number of interactions yields the best performance. In particular, we compare numbers {0, 1, 2, 3, 4, 5}. It can be seen from Fig. 8 that three times of interaction step achieves the best performance among all the compared numbers. In addition, the result of number 2 is also satisfactory, indicating that we can consider 2 interactions if our computational resources are limited. This result demonstrates the robustness of our choice of the number 3 for the interaction step.

Conclusion

In this paper, we propose a document-level relation extraction method-SKAMRR. Instead of simply using entities or words to build graphs, we employ AMR as the basis for building document graphs using sememe-enhanced word representation and interacting with helpful information through the document-level AMR graphs. Afterward, we get the entities' features and build the entity pairs graph for relation reasoning. Finally, we design a global threshold adaptation loss function that can alleviate the problem of unbalanced category samples in the dataset. Experimental results show that SKAMRR achieves very competitive performance in

ule, the sememe knowledge, and the loss function (GAL). Table 3 shows the results. First, we use the (GAIN) scheme instead of AMR to construct the document graph, and we can observe this leads to a 1.37% drop in Ign $F1$ and 1.4% in $F1$. Then, we remove the Sememe information from the model, and it also leads to a decrease of the model by 1.11% on Ign $F1$ and 1.12% on $F1$. Afterwards, We replace the loss function with the conventional adaptive thresholding loss [19], the $F1$ and Ign $F1$ decreased by 1.14% and 1.18%. Finally, we remove all three components mentioned above, and the experimental metrics dropped even more, with a 1.93% drop in Ign $F1$ and a 2.07% drop in $F1$. The ablation experiments can demonstrate that the building block of the AMR graph structure used in this paper plays a key role, and the effect of the model decreases the most if this part is removed. The above experiments also show the effectiveness of sememe knowledge fusion, and loss function (GAL) that we used, and the experimental results depend on each part of the method.

In the second part, we create the long-tail data in the dataset, containing 86 relations. The experiment is to verify the validity of GAL proposed in this paper. The results of the experiment are shown in Table 4. We can see that the $F1$ value of our model decreases by 0.91% and 0.65% when we use normal CE and ATL. This proves that the GAL proposed
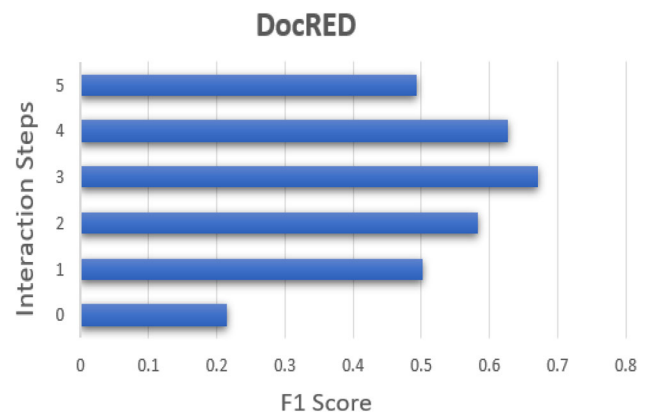
both real-world datasets, which verified its effectiveness. For future work, (1) use graph comparison learning to improve the performance of document-level relationship extraction tasks based on AMR graphs; (2) design a unified framework that unifies the graph interaction process at different stages, so that both the interaction purpose and the computational complexity can be achieved; (3) continue to explore new loss functions to better solve the problem of uneven data distribution that exists in the dataset (long-tail data).

Data availability All datasets used are publically available and the sources are properly cited in the paper.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Liu Shen (2022) Aspect term extraction via information-augmented neural network. Complex Intell Syst. https://doi.org/10.1007/s40747-022-00818-2
2. Tang et al (2022) Attensy-sner software knowledge entity extraction with syntactic features and semantic augmentation information. Complex Intell Syst. https://doi.org/10.1007/s40747-022-00742-5
3. Wei Z et al (2020) A novel cascade binary tagging framework for relational triple extraction. In: Proceedings of the 58th annual meeting of the association for computational linguistics. Association for Computational Linguistics, pp 1476–1488 **(online)**
4. Wang H et al (2019) Extracting multiple-relations in one-pass with pre-trained transformers. In: Proceedings of the 57th annual meeting of the association for computational linguistics. Association for Computational Linguistics, Florence, Italy, pp 1371–1377
5. Zhou P et al (2016) Attention-based bidirectional long short-term memory networks for relation classification. In: Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: short papers). Association for Computational Linguistics, Berlin, Germany, pp 207–212
6. Yao Y et al (2019) DocRED: a large-scale document-level relation extraction dataset. In: Proceedings of the 57th annual meeting

of the Association for Computational Linguistics. Association for Computational Linguistics, Florence, Italy, pp 764–777
7. Christopoulou F et al (2019) Connecting the dots: Document-level neural relation extraction with edge-oriented graphs. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP). Association for Computational Linguistics, Hong Kong, China, pp 4925–4936
8. Zeng S et al (2020) Double graph based reasoning for document-level relation extraction. In: Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP) Association for Computational Linguistics, pp 1630–1640 **(online)**
9. Xu W et al (2021) Discriminative reasoning for document-level relation extraction. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. Association for Computational Linguistics, pp 1653–1663 **(online)**
10. Nan G et al (2020) Reasoning with latent structure refinement for document-level relation extraction. In: Proceedings of the 58th annual meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp 1546–155 **(online)**
11. Xu B et al (2021) Entity structure within and throughout: modeling mention dependencies for document-level relation extraction. In: Thirty-fifth AAAI conference on artificial intelligence, AAAI 2021. AAAI Press, pp 14149–14157 **(online)**
12. Zhang N et al (2021) Document-level relation extraction as semantic segmentation. In: Zhou Z (ed) Proceedings of the thirtieth international joint conference on artificial intelligence, IJCAI 2021. ijcai.org, Montreal, Canada, pp 3999–4006
13. Xu K et al (2019) How powerful are graph neural networks? In: 7th international conference on learning representations, ICLR 2019, New Orleans, LA, USA, May 6–9, 2019. OpenReview.net, New Orleans, LA, USA
14. Quirk C (2017) Hoifung: distant supervision for relation extraction beyond the sentence boundary. In: Proceedings of the 15th conference of the european chapter of the Association for Computational Linguistics: volume 1, long papers. Association for Computational Linguistics, Valencia, Spain, pp 1171–1182
15. Xue Z, Li R, Dai Q, Jiang Z (2022) Corefdre: document-level relation extraction with coreference resolution. arXiv:2202.10744
16. Wang D et al (2020) Global-to-local neural networks for document-level relation extraction. In: Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP). Association for Computational Linguistics, pp 3711–3721 **(online)**
17. Li B et al (2020) Graph enhanced dual attention network for document-level relation extraction. In: Proceedings of the 28th international conference on computational linguistics. International Committee on Computational Linguistics, Barcelona, Spain, pp 1551–1560 **(online)**
18. Zhang Z et al (2020) Document-level relation extraction with dual-tier heterogeneous graph. In: Proceedings of the 28th international conference on computational linguistics. International Committee on Computational Linguistics, Barcelona, Spain, pp 1630–1641 **(online)**
19. Xu W et al (2021) Document-level relation extraction with reconstruction. In: Thirty-fifth AAAI conference on artificial intelligence, AAAI 2021. AAAI Press, pp 14167–14175 **(online)**
20. D Li et al (2021) MRN: a locally and globally mention-based reasoning network for document-level relation extraction. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. Association for Computational Linguistics, pp 1359–1370 **(online)**
21. Zeng et al (2021) SIRE: separate intra- and inter-sentential reasoning for document-level relation extraction. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. Association for Computational Linguistics, pp 524–534 **(online)**

22. Ye et al (2020) Coreferential reasoning learning for language representation. In: Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP). Association for Computational Linguistics, pp 7170–7186 **(online)**

23. Zhou et al (2021) Document-level relation extraction with adaptive thresholding and localized context pooling. In: Thirty-fifth AAAI conference on artificial intelligence, AAAI 2021. AAAI Press, pp 14612–14620 **(online)**

24. Wang X, Wang Z, Sun W, Hu W (2022) Enhancing document-level relation extraction by entity knowledge injection. In: The semantic web—ISWC 2022—21st international semantic web conference, virtual event, October 23–27, 2022, Proceedings. lecture notes in computer science, vol 13489. Springer, pp 39–56. https://doi.org/10.1007/978-3-031-19433-7_3D

25. Zhang L, Cheng Y (2022) A densely connected criss-cross attention network for document-level relation extraction. arxiv:2203.13953. https://doi.org/10.48550/2203.13953

26. Eberts M, Ulges A (2021) An end-to-end model for entity-level relation extraction using multi-instance learning. In: Merlo P, Tiedemann J, Tsarfaty R (eds.) Proceedings of the 16th conference of the european chapter of the Association for Computational Linguistics: main volume, EACL 2021, Online, April 19–23, 2021. Association for Computational Linguistics, pp 3650–3660. https://doi.org/10.18653/v1/2021.eacl-main.319

27. Hogan W, Li J, Shang J (2022) Fine-grained contrastive learning for relation extraction. In: Goldberg Y, Kozareva Z, Zhang Y (eds) Proceedings of the 2022 conference on empirical methods in natural language processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7–11, 2022. Association for Computational Linguistics, pp 1083–1095. https://aclanthology.org/2022.emnlp-main.71

28. Du Y, Ma T, Wu L, Wu Y, Zhang X, Long B, Ji S (2022) Improving long tailed document-level relation extraction via easy relation augmentation and contrastive learning. arXiv:2205.10511. https://doi.org/10.48550/2205.10511

29. Kipf TN, Welling M (2017) Semi-supervised classification with graph convolutional networks. In: 5th international conference on learning representations, ICLR 2017. OpenReview.net, Toulon, France. https://openreview.net/forum?id=SJU4ayYgl

30. Velickovic P, Cucurull G, Casanova A, Romero A, Liò P, Bengio Y (2018) Graph attention networks. In: 6th international conference on learning representations, ICLR 2018. OpenReview.net, Vancouver, BC, Canada. https://openreview.net/forum?id=rJXMpikCZ

31. Huang B, Carley KM (2019) Syntax-aware aspect level sentiment classification with graph attention networks. In: Inui K, Jiang J, Ng V, Wan X (eds) Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing, EMNLP-IJCNLP 2019. Association for Computational Linguistics, Hong Kong, China, pp 5468–5476. https://doi.org/10.18653/v1/D19-1549

32. Wang K, Shen W, Yang Y, Quan X, Wang R (2020) Relational graph attention network for aspect-based sentiment analysis. In: Jurafsky D, Chai J, Schluter N, Tetreault JR (eds) Proceedings of the 58th annual meeting of the Association for Computational Linguistics, ACL 2020. Association for Computational Linguistics, pp 3229–3238. https://doi.org/10.18653/v1/2020.acl-main.295 **(online)**

33. Liu Z, Xiong C, Sun M, Liu Z (2020) Fine-grained fact verification with kernel graph attention network. In: Jurafsky D, Chai J, Schluter N, Tetreault JR (eds) Proceedings of the 58th annual meeting of the Association for Computational Linguistics, ACL 2020. Association for Computational Linguistics, pp 7342–7351. https://doi.org/10.18653/v1/2020.acl-main.655 **(online)**

34. Tu M, Wang G, Huang J, Tang Y, He X, Zhou B (2019) Multi-hop reading comprehension across multiple documents by reasoning over heterogeneous graphs. In: Korhonen A, Traum DR, Màrquez L (eds) Proceedings of the 57th conference of the Association for Computational Linguistics, ACL 2019. Association for Computational Linguistics, Florence, Italy, pp 2704–2713. https://doi.org/10.18653/v1/p19-1260

35. Wang D, Liu P, Zheng Y, Qiu X, Huang X (2020) Heterogeneous graph neural networks for extractive document summarization. In: Jurafsky D, Chai J, Schluter N, Tetreault JR (eds) Proceedings of the 58th annual meeting of the Association for Computational Linguistics, ACL 2020. Association for Computational Linguistics, pp 6209–6219. https://doi.org/10.18653/v1/2020.acl-main.553 **(online)**

36. Li W, Xu J, He Y, Yan S, Wu Y, Sun X (2019) Coherent comments generation for Chinese articles with a graph-to-sequence model. In: Korhonen A, Traum DR, Màrquez L (eds) Proceedings of the 57th conference of the Association for Computational Linguistics, ACL 2019. Association for Computational Linguistics, Florence, Italy, pp 4843–4852. https://doi.org/10.18653/v1/p19-1479

37. Lu Y, Li C (2020) GCAN: graph-aware co-attention networks for explainable fake news detection on social media. In: Jurafsky D, Chai J, Schluter N, Tetreault JR (eds) Proceedings of the 58th annual meeting of the Association for Computational Linguistics, ACL 2020. Association for Computational Linguistics, pp 505–514. https://doi.org/10.18653/v1/2020.acl-main.48 **(online)**

38. Li A, Qin Z, Liu R, Yang Y, Li D (2019) Spam review detection with graph convolutional networks. In: Zhu W, Tao D, Cheng X, Cui P, Rundensteiner EA, Carmel D, He Q, Yu JX (eds) Proceedings of the 28th ACM international conference on information and knowledge management, CIKM 2019. ACM, Beijing, China, pp 2703–2711. https://doi.org/10.1145/3357384.3357820

39. Dong (2003) Dong: Hownet—a hybrid language and knowledge resource. In: NLP-KE

40. Liu Y, Zhang M, Ji D (2020) End to end Chinese lexical fusion recognition with sememe knowledge. In: Scott D, Bel N, Zong C (eds) Proceedings of the 28th international conference on computational linguistics, COLING 2020. International Committee on Computational Linguistics, Barcelona, Spain, pp 2935–2946. https://doi.org/10.18653/v1/2020.coling-main.263

41. Hou B, Qi F, Zang Y, Zhang X, Liu Z, Sun M (2020) Try to substitute: An unsupervised chinese word sense disambiguation method based on hownet. In: Scott D, Bel N, Zong C (eds) Proceedings of the 28th international conference on computational linguistics, COLING 2020, Barcelona, Spain (online), December 8-13, 2020. International Committee on Computational Linguistics, Barcelona, Spain, pp 1752–1757. https://doi.org/10.18653/v1/2020.coling-main.155

42. Zang Y, Qi F, Yang C, Liu Z, Zhang M, Liu Q, Sun M (2020) Word-level textual adversarial attacking as combinatorial optimization. In: Jurafsky D, Chai J, Schluter N, Tetreault JR (eds) Proceedings of the 58th annual meeting of the Association for Computational Linguistics, ACL 2020. Association for Computational Linguistics, pp 6066–6080. https://doi.org/10.18653/v1/2020.acl-main.540 **(online)**

43. Qi F et al (2019) Openhownet: an open sememe-based lexical knowledge base. arxiv:1901.09957

44. Banarescu L, Bonial C, Cai S, Georgescu M, Griffitt K, Hermjakob U, Knight K, Koehn P, Palmer M, Schneider N (2013) Abstract meaning representation for sembanking. In: Dipper S, Liakata M, Pareja-Lora A (eds) Proceedings of the 7th linguistic annotation workshop and interoperability with discourse, LAW-ID@ACL 2013. The Association for Computer Linguistics, Sofia, Bulgaria, pp 178–186. https://aclanthology.org/W13-2322/

45. Bai X, Chen Y, Zhang Y (2022) Graph pre-training for AMR parsing and generation. In: Muresan S, Nakov P, Villavicencio A (eds) Proceedings of the 60th annual meeting of the Association for Computational Linguistics (volume 1: long Papers), ACL 2022, Dublin, Ireland, May 22–27, 2022. Association for Computational Linguis-

tics, Dublin, Ireland, pp 6001–6015. https://doi.org/10.18653/v1/2022.acl-long.415

46. Li I, Song L, Xu K, Yu D (2022) Variational graph autoencoding as cheap supervision for AMR coreference resolution. In: Muresan S, Nakov P, Villavicencio A (eds) Proceedings of the 60th annual meeting of the Association for Computational Linguistics (volume 1: long papers), ACL 2022, Dublin, Ireland, May 22–27, 2022. Association for Computational Linguistics, Dublin, Ireland, pp 2790–2800. https://doi.org/10.18653/v1/2022.acl-long.199

47. Shou Z, Jiang Y, Lin F (2022) AMR-DA: data augmentation by abstract meaning representation. In: Muresan S, Nakov P, Villavicencio A (eds) Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22–27, 2022. Association for Computational Linguistics, Dublin, Ireland, pp 3082–3098. https://doi.org/10.18653/v1/2022.findings-acl.244

48. Devlin J, Chang M-W, Lee K, Toutanova K (2019) 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, vol 1 (long and short papers). Association for Computational Linguistics, Minneapolis, Minnesota, pp 4171–4186

49. Luong T, Pham H, Manning CD (2015) Effective approaches to attention-based neural machine translation. In: Proceedings of the 2015 conference on empirical methods in natural language processing, EMNLP 2015. The Association for Computational Linguistics, Lisbon, Portugal, pp 1412–1421

50. Xu D, Li J, Zhu M, Zhang M, Zhou G (2020) Improving AMR parsing with sequence-to-sequence pre-training. In: Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP). Association for Computational Linguistics, pp 2501–2511 **(online)**

51. Schlichtkrull M et al (2018) Modeling relational data with graph convolutional networks. In: Gangemi A, Navigli R, Vidal M, Hitzler P, Troncy R, Hollink L, Tordai A, Alam M (eds) The semantic web—15th international conference, ESWC 2018. Lecture notes in computer science, vol 10843. Springer, Crete, Greece, pp 593–607

52. Li B et al (2019) Gradient harmonized single-stage detector. In: The thirty-third AAAI conference on artificial intelligence, AAAI 2019. AAAI Press, Honolulu, Hawaii, USA, pp 8577–8584

53. Li J et al (2016) Biocreative V CDR task corpus: a resource for chemical disease relation extraction. Database J Biol Databases Curation 2016. https://doi.org/10.1093/database/baw068

54. Wu Y, Luo R, Leung HCM, Ting H, Lam TW (2019) RENET: a deep learning approach for extracting gene-disease associations from literature. In: Cowen LJ (ed) Research in computational molecular biology—23rd annual international conference, RECOMB 2019, Washington, DC, USA, May 5–8, 2019, Proceedings. lecture notes in computer science, vol 11467. Springer, pp 272–284. https://doi.org/10.1007/978-3-030-17083-7_17

55. Paszke A et al (2017) Automatic differentiation in PyTorch. In: 2017 NIPS workshop autodiff, Long Beach, USA, Dec 4–9, 2017. OpenReview.net

56. Wang MY et al (2019) Deep graph library: towards efficient and scalable deep learning on graphs. arXiv:1909.01315

57. Liu Y et al (2019) Roberta: a robustly optimized BERT pretraining approach. arXiv:1907.11692

58. Fernandez Astudillo R, Ballesteros M, Naseem T, Blodgett A, Florian R (2020) Transition-based parsing with stack-transformers. In: Findings of the Association for Computational Linguistics: EMNLP 2020. Association for Computational Linguistics, pp 1001–1007 **(online)**

59. Loshchilov I, Hutter F (2019) Decoupled weight decay regularization. In: 7th international conference on learning representations, ICLR 2019. OpenReview.net, New Orleans, LA, USA

60. Zhou H et al (2020) Global context-enhanced graph convolutional networks for document-level relation extraction. In: Proceedings of the 28th international conference on computational linguistics. International Committee on Computational Linguistics, Barcelona, Spain, pp 5259–5270 **(online)**

61. Zhao C, Zeng D, Xu L, Dai J (2022) Document-level relation extraction with context guided mention integration and inter-pair reasoning. CoRR arXiv:2201.04826

62. Wang H et al (2019) Fine-tune bert for docred with two-step process. CoRR arXiv:1909.11898

63. Tang H et al (2020) HIN: hierarchical inference network for document-level relation extraction. In: 24th Pacific-Asia conference, PAKDD 2020. Lecture notes in computer science, vol 12084. Springer, Singapore, pp 197–209