Self-supervised Enhancement for Named Entity Disambiguation via Multimodal Graph Convolution

Pengfei Zhou[1], Kaining Ying[1], Zhenhua Wang, Dongyan Guo, and Cong Bai*

***Abstract*—Named entity disambiguation (NED) finds the specific meaning of an entity mention in a particular context and links it to a target entity. With the emergence of multimedia, the modalities of content on the Internet have become more diverse, which poses difficulties for traditional NED, and the vast amounts of information make it impossible to manually label every kind of ambiguous data to train a practical NED model. In response to this situation, we present MMGraph, which uses multimodal graph convolution to aggregate visual and contextual language information for accurate entity disambiguation for short texts, and a self-supervised simple triplet network (SimTri) that can learn useful representations in multimodal unlabeled data to enhance the effectiveness of NED models. We evaluated these approaches on a new dataset, MMFi, which contains multimodal supervised data and large amounts of unlabeled data. Our experiments confirm the state-of-the-art performance of MMGraph on two widely used benchmarks and MMFi. SimTri further improves the performance of NED methods. The dataset and code are avaliable at: https://github.com/LanceZPF/NNED_MMGraph.**

***Index Terms*—Named entity disambiguation, multimodal data, self-supervised learning, graph convolutional network.**

I. Introduction

NAMED entity disambiguation (NED) aims to find the substantive corresponding entity for an ambiguous entity mention in a natural context. With the growing demand for accurate content on the Internet, lexical ambiguity makes information transmission inefficient. As shown in Fig. 1, the word "Apple" in a financial report could mean either Apple the company, or, apple, the fruit. However, with few normative tools to process entity disambiguation, manual check is impractical given the volume of data. In this case, NED can help to discriminate the potential ambiguity of a word and accurately link it to a target entity [1], which ensures efficient content delivery. Furthermore, academic research on knowledge graphs (KGs) also requires NED to ensure accurate, unambiguous entity linking [2].

Current NED approaches focus on obtaining an embedding that makes use of the rich context in natural language texts. For example, the local neural attention mechanism proposed by Ganea *et al.* [3] obtains entity embeddings with local context windows. Shahbazi *et al.* [4] used global context information to improve entity disambiguation accuracy in documents. The difficulty in the NED task is that it suffers from insufficient context information in short texts, meaning that it has to rely more on prior information and external knowledge base. Existing methods ignore the use of sentence structure and rely on the use of context information to achieve better contextual embedding. Although K-NED [5] works on external knowledge to obtain the enhanced representation, it still fails to take into account prior knowledge in sentences.

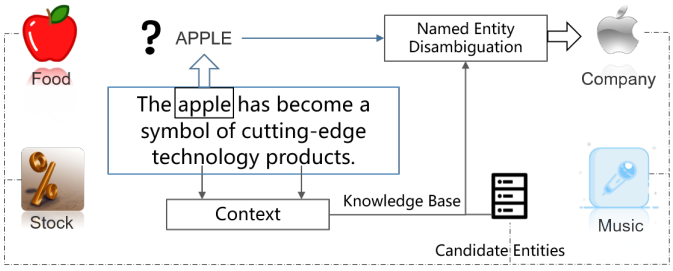


Fig. 1. Named entity disambiguation. When ambiguity is eliminated, the word "Apple" is linked to the exact entity "Apple Inc."

Methods based on plaintext have limited accuracy because they cannot exploit useful multimodal representations to better serve the demand of NED. With the evolution of multimedia technologies, learning representations on multimodal data are attracting much attention because most data in the real world are multimodal [6]. Accordingly, the NED methods that can ensure the accuracy of multimodal information bridging the gap between modalities are to be proposed. The exploration of multimodal NED methods can also promote the development of the multimodal knowledge graph [7]–[9], which is the key to research fields such as intelligent medicine [10].

The vast amount of unlabeled data on the Internet is neglected in NED tasks, because most data from social media such as Twitter lack the necessary information to provide available labels, and the metadata that humans can annotate is just the tip of the iceberg (manual annotation on large-scale data is time-consuming and laborious). Therefore, using large amounts of unlabeled data to get rid of manual annotation is still a technical challenge [11]. Recently, self-supervised learning (SSL) has emerged as a popular topic in unsupervised learning [12], and the research of SSL on unlabeled data can not only improve the efficiency of NED methods but also help

*Corresponding Author
[1]Equal Contribution

This work is supported by Zhejiang Provincial Natural Science Foundation of China under Grant No. LR21F020002 and Natural Science Foundation of China under Grant No. U20A20196.

P. Zhou is with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China, the University of Chinese Academy of Sciences, Beijing 100049, China, and also with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China (e-mail: zpf4wp@outlook.com).

K. Ying, Z. Wang, D. Guo and C. Bai are with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China (e-mail: ykn, zhhwang, guodongyan, congbai@zjut.edu.cn).

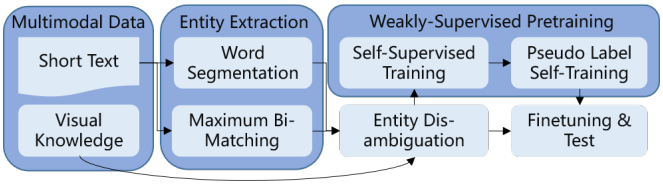


Fig. 2. Our end-to-end entity disambiguation pipeline. Bi-matching refers to bidirectional matching.

people to explore the inner mechanism of human recognition [13], [14]. It is known that people can easily learn to identify ambiguities in unlabeled conversations even for unknown new words [15]. Meanwhile, a functional NED dataset with large-scale unlabeled data is needed for researching on SSL.

To deal with the above problems, we establish a new database, MMFi, built from financial news, with a large amount of unlabeled data from the Internet. The multimodality of visual logos is adopted to enhance the performance of NED. We believe that with abundant information from multimodal data, it is easier to find the pattern of an ambiguous sentence with the help of the external knowledge of aligned multimodal pairs. Therefore, the proposed dataset can help develop multimodal NED methods and facilitate the exploration of self-supervised learning in NED as a new benchmark. As shown in the pipeline in Fig. 2, we also propose an end-to-end entity disambiguation system based on the new NED model. MMGraph employs employing pretrained BERT [16] and an advanced multimodal graph convolutional network (GCN) [17] to obtain multimodal embeddings by fusing context structures and images, so as to improve final entity disambiguation performance by learning better representations. Experiments on two benchmarks and MMFi show that the proposed graph convolution for both text and multimodal data can help improve the accuracy of NED.

With large-scale unlabeled data in new datasets, we present a self-supervised pipeline, Simple Triplet Network (SimTri). The proposed SimTri can learn useful representation from unlabeled data using latent information and unique characteristics, such as the aligned cross-modal feature from multimodal data, and is fine-tuned on a small amount of supervised data to achieve better NED performance.

The contributions of this study can be summarized as follows: 1) We establish MMFi, which contains a large amount of unlabeled multimodal data, and is appropriate for evaluating NED algorithms with respect to supervised and self-supervised learning; 2) We propose an approach that adapts to both traditional and multimodal NED tasks. MMGraph can use the prior sentence structure and correlation of multimodal data to achieve accurate entity disambiguation; and 3) The self-supervised SimTri framework is proposed to exploit unlabeled multimodal data and improve NED performance with the architecture of a triplet network.

The rest of this paper is organized as follows. Related work is discussed in Section II. The construction of the MMFi dataset is discussed in Section III. Section IV presents the supervised MMGraph method. We discuss the self-supervised SimTri in Section V. Experiments and results are detailed in Section VI. We conclude with a discussion of the contribution of our work in Section VII.

II. Related Work

A. Named Entity Disambiguation

A named entity with ambiguity can refer to multiple conceptual entities, which named entity disambiguation (NED) aims to eliminate with the help of context or an external knowledge base [18]. Traditional NED approaches are mostly based on the semantic similarity between mentions and candidate entities [19]. Early work [20] focused on the metrics of measurement. With the development of deep learning, NED approaches are now based on neural networks, and they can obtain better embeddings for mentions, entities, and contexts. Kolitsas *et al.* [21] developed an end-to-end entity disambiguation system that could analyze context based on a neural network with attention and a global voting mechanism. Sevgili *et al.* [22] embedded a neural NED model in an end-to-end entity disambiguation system to verify the practicability of NED methods.

Previous methods adopted shallow models such as Word2Vec [23] and Doc2Vec [24]. Some work [25] used deeper networks such as CNN, but achieved only limited performance in this NLP task. Deep pretrained models such as BERT [16] show enormous potential in various NLP fields, including NED. For example, the model proposed by Broscheit [26] can achieve accurate disambiguation with the help of pretrained BERT [16].

The knowledge graph has attracted much attention in data mining [27]. The use of knowledge graphs as data structures has also gained the attention of NED researchers [2]. For example, the graph embedding method [22] uses the external information of candidate entities and learns useful embeddings. Such designs can take advantage of the structure of a database. However, this method is based on DeepWalk [28]. Deep networks such as the GCN are used in the NED area to obtain a better representation for entity and context. The multimodal knowledge graph is also attracting interest due to its rich information modeling and similarities to real-world environments. However, multimodal entity disambiguation [29] is still under development.

B. Graph Neural Networks

In the field of graph learning, Gori *et al.* [30] first propose the idea of graph neural network (GNN), and Scarselli *et al.* [17] sought to improve the performance of graph learning. Kipf *et al.* [31] proposed the graph convolutional network (GCN), which was a landmark contribution to graph learning. Researchers have subsequently explored the performance improvement in graph learning. For example, Velickovic *et al.* [32] proposed a graph attention network to integrate the attention mechanism in a GNN. GraphSAGE [33] uses graph segmentation techniques to improve the practicability of GNNs, and it can perform graph convolution on a much larger graph.

The core idea of graph learning is to learn a mapping function, through which the nodes in the graph can aggregate

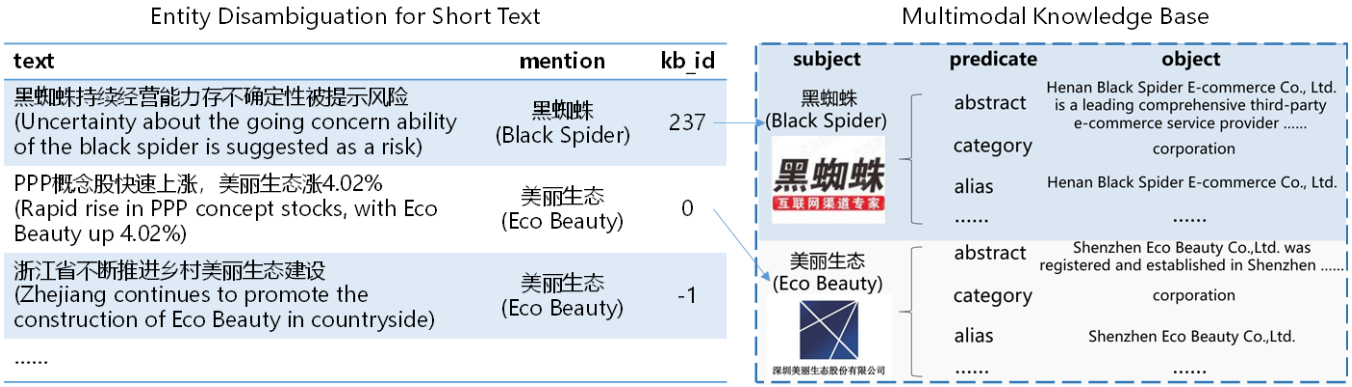


Fig. 3. Established MMFi. Annotation of an entity mention is linked to a corresponding entity in the knowledge base by the label kb_id, which is the index of a knowledge base item, and kb_id= –1 indicates that no corresponding entities exist in the knowledge base.

their features with those of their neighbors to generate a new node representation. It enables GCN to extend the convolution operation of traditional data (such as images) to graphic data (such as feature graphs). Based on the contribution of graph learning and applications in data mining, GNNs also shows potential in computer vision, natural language processing (NLP) and multimedia tasks such as visual tracking [34], semantic role labeling [35], group activity recognition [36], and image caption [37]. An example is the multimodal graph learning in CMRDF [38] explores the deep semantic correlation of each modality. It is observed that the GNNs can deal with the potential correlation of features, which can be complex in real data, and use these associations to help align the data in different modalities. As a result, GCN has been chosen to realize multimodal entity disambiguation, bridging semantic gaps between visual objects and natural language entities in our framework.

C. Self-Supervised Learning

Self-supervised learning (SSL) is a much-anticipated direction of unsupervised learning [39]. In the early years, self-supervised approaches based on autoencoders have been used to learn latent representations from datasets as image/text representations [40]. The auxiliary augmentation of training data generally becomes a basic solution in SSL, such as the use of colorization [41] to augment data and train a model to generate a color map from greyscale. Several SSL methods in computer vision are inspired by the self-supervised BERT model [16], which uses a cloze test on sentences to train on a large unlabeled corpus, and achieves state-of-the-art performance for multiple NLP tasks. GPT-3 [15] uses a larger model architecture and corpus to similarly train a more powerful model. Other such applications include the image jigsaw puzzle [42], image inpainting [43] and the relative patch prediction [44], [45].

Methods employing contrastive learning [46] based on Siamese networks [47] and contrastive loss [48], [49] have eclipsed the above work. MOCO proposed by He *et al.* [50] employs a memory bank with momentum updates. SimCLR *et al.* [14] achieves promising performance via large batch size training. BYOL [51] and SimSiam [52] further use stop-grad for more efficient self-supervised training. These methods are all based on the idea of training models by the maximum similarity of representations from the same class. By training on unlabeled data in a contrastive task-agnostic way, the model can learn a better data representation and perform better in downstream tasks.

Vitality in SSL is also attracting the attention of multimedia researchers [53]. Previous work focused on time and multimodal information from videos to train sequential models [54], [55], relying on the time sequential signal by predicting shuffled frames [56], the direction of time [57] and sequence order [58]. However, there has been limited research on multimodal data of plain images and texts, as the work is arduous and the data scant [59]. SSL shows the potential to solve weakly-supervised problems [60]. Hence, it is proving to be a valuable research topic to address our unlabeled data problem in entity disambiguation datasets with new SSL frameworks.

III. Dataset

In this section, we describe the data collection, arrangement, and processing of MMFi.

A. Data Collection

The metadata of our database consist of short texts with financial entities taken from financial news downloaded from Chinese financial websites. websites[1][2]. For example, “Intercontinental Hotel of Financial Street turns into an office building” has the mention of “Financial Street” to be disambiguated, because the entity “Financial Street” can refer to either a road or a registered company. The original dataset is partly annotated with entity mentions and corresponding meanings. Furthermore, the meta-dataset leaves a larger number of unlabeled raw sentences than labeled data, and we define missing labels in the dataset as the noise of data.

Unlike long-text NED tasks [4], [61], the short-text NED task does not have rich context, which makes the task more

[1]http://www.10jqka.com.cn/
[2]https://www.hundsun.com/summit.aspx

difficult due to the lack of background information. Therefore, external knowledge should offer more abundant information to achieve accurate NED results. The construction of our knowledge base depends on the Baidu encyclopedia, Wiki encyclopedia, and data mining tools. No companies in our dataset share the same name. However, these entities can be roughly classified into two categories: financial entities as a positive sample and non-financial entities as a negative sample. Non-financial entities are obtained from fields such as food and medicine. Therefore, our NED task aims to identify whether a mention refers to a corporation entity or an entity from a vertical field. We collect the contents of related items with the mentioned entity words in the dataset and filter out extraneous data using pretrained BERT [16], which is fine-tuned on a two-class mini-dataset to distinguish noisy data. We perform a manual recheck to ensure data quality.

We collect images by Google API to enhance our knowledge base with respect to multimedia. The images are roughly cleaned by the classic random forest method and double-checked manually. Positive samples are mostly company logos of financial entities and negative samples correspond to potential ambiguous entities in the knowledge base. These image data can be used to enhance the research on entity disambiguation methods.

B. Data Arrangement

After collection, supplementation, and cleaning, we arrange these metadata in dictionary form. The established external knowledge base has 830 multimodal pairs, and each multimodal pair corresponds to an entity. Each pair has a subject-predicate-object (S-P-O) triplet and a corresponding image. The training/test text data and corresponding entity in the knowledge base are shown in Fig. 3.

The text data in the training set are annotated with index, entity mention, and disambiguated entities corresponding to the knowledge base, whose elements, as shown in Fig. 3, are arranged in S-P-O triplets by the entities corresponding to the training set with ambiguations of entity mention words. These elements are entity_id, entity_name, kd_id number, img_id number, and entity_description. The img_id number refers to the visual image of a particular entity. The entity_description includes the abstract of entity meaning, category of entity, full name or alias, and six stock numbers. With the structure of the designed knowledge base, abundant information is provided for the downstream disambiguation task.

C. Data Processing

The language of the established dataset is Chinese, the text of which must be segmented into phrases. Hence, we require a powerful word segmentation (WS) system for early preprocessing. According to the requirements of downstream modules, the functional analysis of WS schemes has three aspects: 1) provide the basic WS function that achieves the best trade-off between precision and speed; 2) based on the realization of WS, accurately extract Chinese characters of the entity to determine whether a keyword and obtain its offset for the further development of the entity-extraction algorithm; and 3) has semantic analysis functions with suitable granularity and high extendibility for the development of a graph structure in the downstream disambiguation model, so as to facilitate the self-supervised learning with the NED model.

TABLE I
COMPARISON BETWEEN DIFFERENT WORD SEGMENTATION (WS) SYSTEMS.

WS Systems	JIEBA[3]	PYLTP [62]	NLPIR [63]	THULAC [64]
Precision	0.972	0.978	0.973	0.972
Speed	1427.01KB/s	124.36KB/s	426.51KB/s	879.59KB/s
Granularity	medium	fine-grained	fine-grained	medium
Extendibility	high	medium	low	medium

As presented in TABLE I, the existing open-source WS systems provided by institutes are tested on MMFi. The speed comparison shows that JIEBA[3] is the most efficient WS system, even compared with the lightweight version of THULAC [64]. Simultaneously, JIEBA demonstrates the stable segmentation performance on our MMFi, which is a relatively precise dataset with short texts. Through the integrated analysis, we choose JIEBA system as the WS module in our NED pipeline for its high WS efficiency, appropriate segmentation granularity, and high extendibility. Although JIEBA does not have the original part-of-speech tagging function, it can be extended by manipulating the open-source code to merge with the part-of-speech tagging function provided by NLPIR [63]. Thus, JIEBA becomes the best choice to be integrated into our end-to-end system. The NLPIR system, which has the advantage of small analytic granularity and a higher speed than PYLTP [62], is also employed for data analysis of the proposed knowledge base.

The pipeline of our processing schemes is described as follows: 1) extract the implied features of the entity to be disambiguated in the sentence (through the analysis of part of speech, offset, and context); 2) analyze the sentence structure and semantic features of the passage and indicate the grammar characteristic of words (through the semantic roles and sentence structure); 3) locate the Chinese character (entity mention to be disambiguated) of the entity as the center, and take the context words before and after it to accumulate source materials to design the graph structure of the core entity disambiguation algorithm; and 4) provide a feasible scheme for using the global context, such as the correlation between sentences with the characteristics of the entities.

Finally, the processed dataset has annotated labels that indicate whether a mention is truly a corporation entity and an index numbers linking to the corresponding external knowledge in the knowledge base. As mentioned above, the external knowledge includes the entity name with abstract, entity description with long context and corresponding images, and all these external knowledge can be input into the NED model together with the sentence so as to conduct disambiguation.

[3]https://github. com/fxsjy/jieba

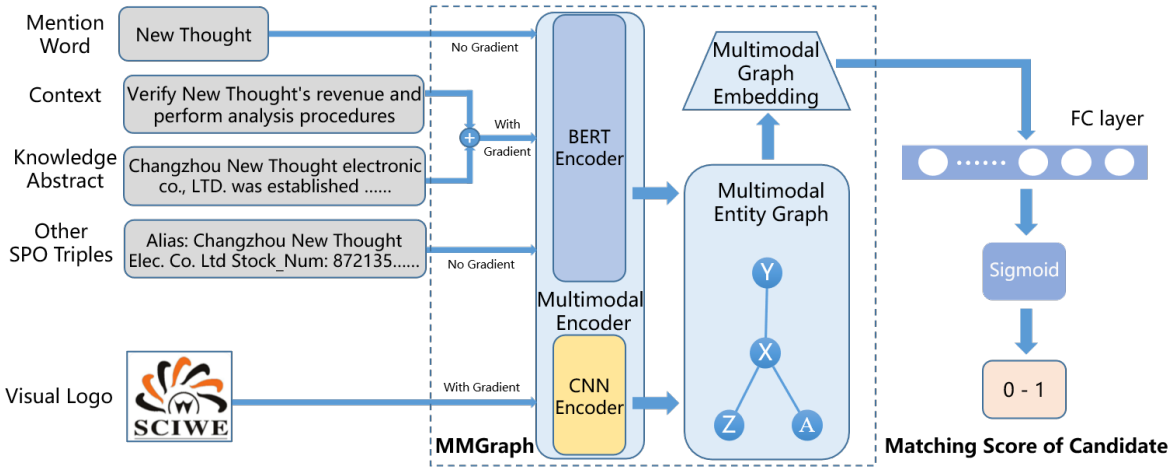


Fig. 4. MMGraph framework. Nodes X, Y, Z, and A represent subgraphs in the global graph structure. X includes the entity and its context, Y denotes knowledge base information corresponding to entity mentions, Z consists of the five words before and after the mention, and A can supplement sentence structure information. With gradient: training with backpropagation; no gradient: training with freezing weights.

IV. GCN-Based Supervised Entity Disambiguation

We provide details of MMGraph in terms of NED in traditional single-text modality and multiple modalities.

A. Entity Extraction

The training and test data in the NED datasets are plaintexts that are not annotated with the mentions that need to be disambiguated. Hence, we need an entity-extraction algorithm to find the candidate words in short texts before NED. We apply the maximum bidirectional matching (bi-matching) algorithm to search for potential mention words that are to be disambiguated. The algorithm searches from left to right and from right to left in the meantime. Entities are formatted into the dictionary tree to accelerate the matching in knowledge base. The JIEBA segmentation library is combined with bi-matching to improve the efficiency of entity extraction.

B. Proposed MMGraph

MMGraph, as shown in Fig. 4, is proposed for supervised NED tasks based on prior graph structures to achieve accurate NED performance without the need for rich word meaning and long contexts. It can exploit multimodal graph embeddings to improve the accuracy of entity disambiguation. We introduce the feature extractor and graph embedding in the text modality, and discuss multimodal entity disambiguation theory.

1) Feature Extractor for Text Embeddings: The intuition of MMGraph relies on pretrained BERT [16], which is combined with a conditional random field (CRF) to form BERT-CRF as a backbone. BERT-CRF works as the feature extractor in our NED approach, and it can provide initial text embeddings based on the context of a mention sentence and the probability distribution in the global NED dataset. This feature extraction is described in what follows.

The proposed disambiguation model includes tokenizer, pre-embedding, Transformer [65] and CRF. Let the length of context for the target word ω is T, the contextual Chinese word sequence is converted into the corresponding vector sequence $[h_1, ..., h_{t-1}, h_t, h_{t+1}..., h_T]$, where h_t is the one-hot representation of the tth word in ω. We use M as the pre-trained word embedding matrix to get the word vector $h'_t = Mh_t$ as the output of the pre-embedding module. Then the context vector c is used to represent the semantic information of the target word context. Similar to the context representation module, the joint representation module uses BERT-CRF to encode the semantic information of definitions, example sentences, and the external knowledge of target words by combining abstract and other S-P-O triples in the dictionary as the input. For the tth word in ω, the example abstract s_t and other S-P-O items g_t form the $x_t = [g_t \# s_t]$, and BERT-CRF is used to integrate dictionary information, using self-attention mechanism to model the semantic information between context vector c and joint vector x. Then, through two layers of graph convolution, multimodal information distribution is integrated into each node via graph embedding.

To be more specific, the input of BERT-CRF is the sum of word embeddings and position embeddings [65]. Then, the multi-head self-attention mechanism of Transformer is used to model the semantic relationship between the context vector and the joint vectors of S-P-O triples, so as to extract the necessary context information. The main components of the Transformer, self-attention layers, calculate the weighted sum of values, where the weight assigned to a value is determined by the correlation between the query and key. In our model, the keys, queries, and values are obtained by linear transformation from the same sequence, which is the joint vector x. If A is the map function of a self-attention layer and x is the input, the output of this layer is

$$A(x) = softmax(\frac{LW_Q(LW_K)^T}{\sqrt{c_k}})(LW_V), \quad (1)$$

where $L \in R^{n \times c}$ represents the input sequence; n, c represent the length and the channels, respectively; $W_Q \in R^{c \times c_k}$, $W_K \in R^{c \times c_k}$ and $W_V \in R^{c \times c_k}$ are respective linear transformations of the query, key, and value matrix; and c_k is the channel

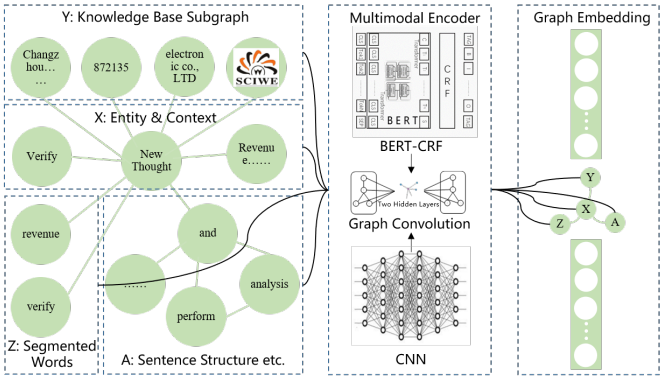


Fig. 5. Multimodal graph with embedding procedures. Nodes are divided into subgraphs according to different attributes.

of the key, where $\sqrt{c_k}$ is used to offset the larger dot product between the query and key. The multi-head mechanism is used to obtain representation from a single input sequence x. For the ith head belonging to h heads, the weight matrices are $W_Q^i \in R^{c\times\frac{c_K}{h}}$, $W_K^i \in R^{c\times\frac{c_K}{h}}$, and $W_V^i \in R^{c\times\frac{c_V}{h}}$, and the multi-head self-attention is

$$MultiHead(x) = Concat(head_1, ..., head_h)W^O, \quad (2)$$

where $W^O \in R^{c\times c}$, and h is set to 8 in this study. To address the vanishing-gradient problem in deep learning [66], the images are normalized by standard batch normalization (BN) in preprocessing, and we apply layer normalization (LN) among the Transformer modules.

Because multi-head self-attention layers consider only long-term context information regardless of dependencies between global labels, the global label distribution is modeled by CRF, and the optimal global label sequence is obtained by considering the adjacency relationship between labels. Then, the latent information of prior syntactic structure is embedded in feature vectors by graph convolution following the setting of adjacency matrices.

2) Graph Embedding via Graph Convolution: Based on BERT-CRF, we propose a graph structure to represent embeddings, which is combined with GCN for more accurate disambiguation. As described earlier, BERT-CRF is used for word segmentation, text embedding, and feature extraction of each element. Our proposed graph convolution uses prior knowledge of context and sentence structure to integrate non-Euclidean spatial information in order to maximize the efficiency of learning graph embeddings via the graph structure. Fig. 5 shows the proposed graph structure and corresponding graph embedding schemes.

Subgraphs in Fig. 5 are simplified as nodes, where X represents an entity mention and its context, Y includes corresponding S-P-O triples in the knowledge base, A supplements sentence structure information from word segmentation, and the five words before and after a mention are integrated into Z. Graph convolution utilizes the graph structure to propagate neighbor node information and learn representations of word embeddings and multimodal features.

We can model the complex interrelationships between nodes through the above graph structure. The purpose of graph learning is to make the correlation between samples with attributes as sparse as possible, and to make the correlation between samples with the same attributes as close as possible. Thus, the relationship between vectors can maintain the structural relationship between nodes in the graph structure. If the mapping function of graph convolution is $G(\cdot)$; then, the input is $I^{i+1} \in R^{n\times d}$ and corresponding correlation matrix is $C \in R^{n\times n}$ (where n is the number of nodes and d is the characteristic dimension). Graph convolution updates layer node properties as $I^{i+1} \in R^{n\times d'}$ by the convolution $I^{i+1} = G(I^i, C)$, or more specifically, it can be rewritten as $I^{i+1} = H(\hat{C}\cdot I^i \cdot W^i)$, where $\hat{C}$ is the normalized version of the correlation matrix and $W^i \in R^{d\times d'}$ is a transformation matrix, and $H(\cdot)$ denotes a non-linear operation of LeakyReLU [67]. Therefore, the proposed graph convolution can be regarded as a graph embedding process. After graph convolution, it is easier to embed the graph in related semantic feature vectors.

MMGraph combines BERT and GCN. We use BERT-CRF backbone to obtain the feature vector of text representation. Let $t \in R^{C\times r}$, where r is the size of the last hidden layer and is set to 768 in our model. Then we obtain the final graph embedding via two graph convolution layers and, based on the mapping function, we obtain $Q = \{t_i\}_{i=1}^{C}$, where C represents the number of the positive vector. We set up the GCN module using the same method mentioned above. The output matrix $Q \in R^{C\times e}$ and e is the size of the label word. The result of graph embedding is a set of the semantic feature vectors in text form, $Z = [z_1, z_2..z_n]$.

The goal of graph-based embedding is to map feature vectors into semantic space. In this way, the solution of the regularization class prediction map can be converted to a sample feature space mapping problem in which X^T is embedded in the semantic space X. If the mapping matrix is M, the corresponding embedding process is expressed as $X = MX^T$. At the same time, M is expressed as

$$\begin{aligned} M &= (XL_CX^T + \lambda X^TX + \eta I)^{-1}(\lambda XY) \\ &= (XL_CX^T + \lambda X^TX + \eta I)^{-1}(\lambda X(2C-1)), \end{aligned} \quad (3)$$

where λ and η control the constraint of graph embedding as the positive role parameter, and L_C is a symmetric and positive semidefinite Laplace matrix. Therefore, the solution of the above equation can be transformed to that of a regularized least squares problem, and the partial derivative of the projection matrix can be solved for. The solution process is

$$M^TXL_CX^TM + \lambda(X^TX - XY) + \eta M = 0. \quad (4)$$

The input sample x with unknown attribute is mapped to semantic space through projection matrix M as

$$y = sign(Mx^T), \quad (5)$$

$$sign(x) = \begin{cases} 1, & if\ x \geq \tau \\ 0, & if\ x < \tau \end{cases}, \quad (6)$$

where τ is an artificial threshold, set to τ = 0.4 by finding the optimal value of our model. We generally use F to extract semantic features. Eventually, the semantic characteristics of

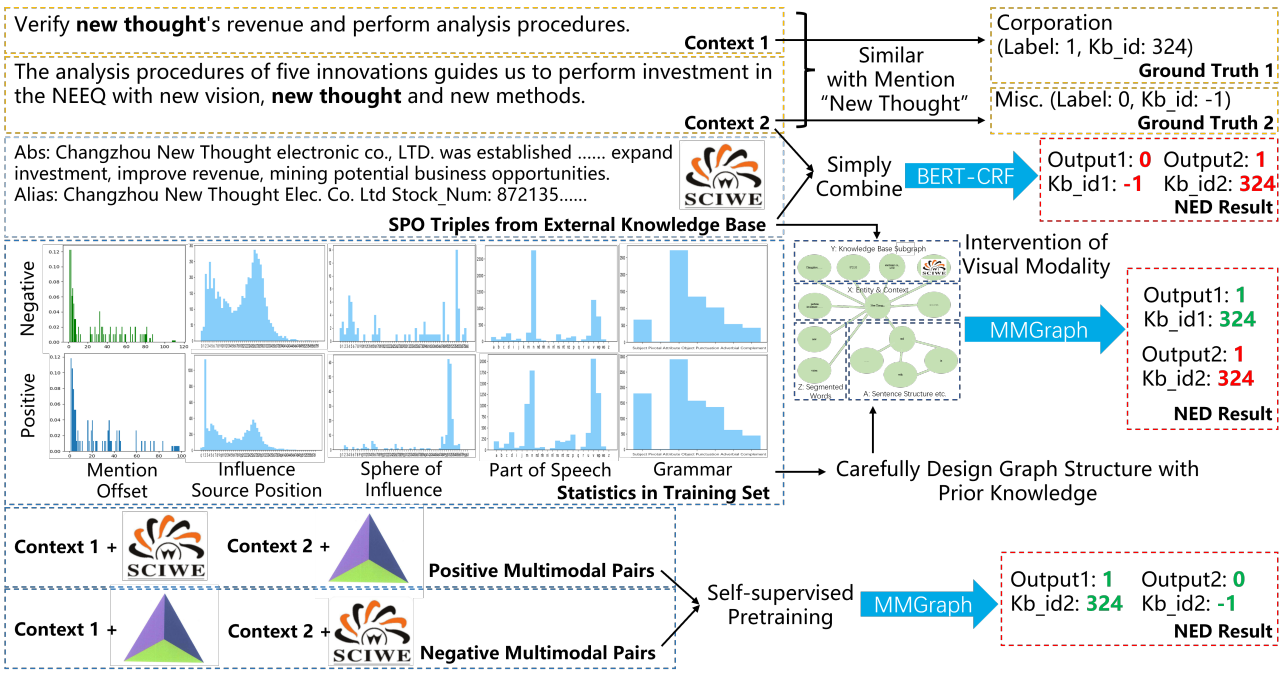


Fig. 6. An illustrative example showing how the proposed MMGraph and the SSL enhancement are theoretically designed. The pipeline of the baseline BERT-CRF is provided for comparison. The label of 0 indicates the negative entities that are not included in the knowledge base, while the label of 0 indicates the positive entities that are included in the knowledge base. The wrong NED results are in red, and the correct predictions are in green color.

the matrix can be $\{y_i\}_{i=1}^{N}$, where $N \leq C$. After the procedures of tokenizer, pre-embedding, Transformer, CRF, graph convolution, fully connected layer, and sigmoid, the loss of model is finally defined as

$$L = \sum_{c=1}^{C} y^c log(\sigma(\hat{y}^c)) + (1 - y^c) log(1 - \sigma(\hat{y}^c)). \quad (7)$$

3) Multimodal Entity Disambiguation: ResNet-50 [66] is used as the pretrained backbone model placed in the GCN to extract visual features. The ResNet module is pretrained at visual logo image classification and integrated in the GCN. We use the extracted visual feature as a node in the subgraph Y of the proposed graph structure, as shown in Fig. 5. The GCN can conduct multimodal graph embedding on this new multimodal graph.

In this multimodal learning task, the final learning results are a set of semantic feature vectors $S = [s_1, s_2..s_n]$. To achieve this, we first learn the concept of graph embedding from the label representation through a stacked GCN with two graph convolution layers, the first with input $P \in R^{C \times D}$, in which the feature vector of entity $x \in R^D$ is extracted by BERT-CRF, where C is the number of nodes in the graph, and D is the dimension of the node vector, which is set to 768 in this paper. The weight is a matrix $W \in R^{C \times d}$, where d is the dimension of the word embedding vector, which is also 768. We can apply a pretrained image classifier ResNet to obtain the visual feature,

$$y = ResNet(v), \quad (8)$$

where v denotes the corresponding visual image in the knowledge base. The visual feature vector $y \in R^D$ represents the second modality data for multimodal graph convolution. The multimodal graph method refers to the subspace learning method [68], and maps the data to the same subspace to reduce the gap between different modalities. The proposed methods transform x and y to the lower-dimension x' and y', respectively, by a linear transformation. The correlation coefficient is used for correlation analysis to find linear transformations corresponding to two groups of variables, ωx and ωy (equal to the dimension of x and y, respectively). Hence, the correlation coefficient between the two variables can be maximized after linear transformation. The algorithm process is as follows. Assuming that the two groups of samples each have N random variables, the linear transformation of these two groups of data can be obtained as follows:

$$z_x \omega_x = (\omega_x^T x_1, ..., \omega_x^T x_N), \quad (9)$$

$$z_y \omega_y = (\omega_y^T x_1, ..., \omega_y^T x_N). \quad (10)$$

The maximum correlation between the two sets of data can be expressed as

$$\rho = \max_{\omega_x, \omega_y} corr(z_x \omega_x, z_y \omega_y) = \max_{\omega_x, \omega_y} \frac{< z_x \omega_x, z_y \omega_y >}{\|z_x \omega_x\| \, \|z_y \omega_y\|}. \quad (11)$$

According to the solution of [69], we can obtain that

$$w_y = \frac{C_{yy}^{-1}}{\lambda} C_{xy} C_{yy}^{-1} C_{yx} \omega_x = \lambda^2 C_{xx} \omega_x. \quad (12)$$

Because the covariance matrices C_{xx} and C_{yy} are positive symmetric, a complete Choleskey decomposition can be performed. Let $\mu_x = R'_{xx} \cdot W_x$. Then

$$R_{xx}^{-1} C_{xy} C_{yy}^{-1} C_{yx} R_{xx}^{-1} u_x = \mu^2 u_x. \quad (13)$$

This is an eigenvalue problem, so we can obtain eigenvector u_x and final W_x by solving $z_x\omega_x$, $z_y\omega_y$, respectively. Furthermore, we obtain embedding function $\rho = corr(X, Y)$ employing (11). We can use ρ to map data from different modalities to the same embedding space. Then we employ the graph embedding function and obtain $S = \{s_i\}_{i=1}^{N}$ to represent the final semantic vector. S is input to the fully connected layer, and the sigmoid function is used to determine the ambiguity category of the embedded output words as a binary classification problem. The loss of the model is the same as Equ. (7).

C. Discussion

To better understand the pipeline throughout the methods of this paper, an example that illustrates how the proposed methods are theoretically designed is shown in Fig. 6. The context instances in the figure are obtained from the MMFi. We can see that the two sentences have the same mention of "new thought" that needs to be disambiguated. However, the lack of abundant context makes the disambiguation on these two sentences difficult. Thus, we require an external knowledge base to provide the basic information (S-P-O triples) to help the model analyze the ambiguity. In this case, BERT-CRF simply combines the sentence with S-P- O triples from the knowledge base as the input of the model. We can see that BERT-CRF fails to disambiguate the mentions in two instances due to the ambiguities in these two sentences, where the context of the two sentences are both similar to the abstract from the external knowledge base.

To deal with these hard cases, we carefully design a graph structure with prior knowledge after conducting comprehensive statistics via NLPIR [63]. It is shown that the positive samples (mentions that refer to the corporation in MMFi) contain fixed prior patterns compared with the negative samples (mentions that refer to the other entities in MMFi). For example, the offset distribution of positive mentions is more concentrated in the front, while that of negative mentions is more even. The mentions of positive samples are more likely to be influenced by words near the beginning of the sentence, and the mentions of positive samples tend to affect the context at the end of the sentence rather than at the beginning of the sentence. Moreover, the patterns in the part-of-the-speech and grammar in positive samples also help design the subgraph of sentence structure. For instance, we make sure that the subject is correlated to the attribute in the graph. MMGraph adopts such a design of the prior graph structure, and the proposed multimodal graph embedding bridges the cross-modal gap after the intervention of the visual modality. Thus, the proposed MMGraph obtains better NED results. However, MMGraph fails on disambiguating the mention in context 2 despite the prior graph structure and multimodal information. We believe the pretraining of self-supervised can further help the model learn the multimodal representations and bring more stable NED performance. Finally, MMGraph can achieve better NED performance with the enhancement of SSL.

Based on the proposed NED methods, we further need an end-to-end entity disambiguation system to directly turn sentences with mention words into disambiguated results with the function of entity linking to the corresponding knowledge base. This system can evaluate the performance of NED at end-to-end entity disambiguation, and it can work as an NLP application to practically process and analyze massive content. Thus, we develop an end-to-end entity disambiguation system based on MMGraph, whose development has four steps:

1) Building the knowledge base (maintenance of external knowledge, including candidate entity attributes). We use the established MMFi database as the knowledge base, and enhance it to ensure its accuracy and richness.

2) Entity extraction (i.e., entity recognition). We implement the JIEBA word segmentation system combined with a maximum bidirectional matching algorithm for the efficient extraction of downstream entity disambiguation. We integrate these segmented materials into the graph structure and model prior knowledge based on the context semantic information.

3) Entity linking (eliminating ambiguity and associating with the knowledge base). BERT-CRF and ResNet are used to extract the semantic vectors for all nodes in a graph, and we obtain the graph embeddings for entity disambiguation through multimodal graph convolution. After training, MMGraph is obtained to build a precise entity disambiguation system. We combine MMGraph with the knowledge base to develop an entity linking module, which finally links the disambiguated word to entity candidates in the knowledge base.

Finally, the end-to-end procedures of entity disambiguation are achieved, and the system is capable of disambiguating texts with massive entity information.

V. Improving Entity Disambiguation with Self-Supervised Learning

We outline the proposed SimTri SSL framework for NED. We implement MMGraph in this framework and utilize the large-scale unlabeled data to further improve the performance of the proposed approach in the multimodal NED task.

A. A Naive SSL Framework for multimodal NED

Labeled training data in the multimedia data from the Web are insufficient for the training of deep neural networks, and there is a risk of overfitting when training complex models. A large amount of unlabeled data on the Internet can be used to learn useful representations. Therefore, we use SSL technology to pretrain unlabeled data and fine-tune a labeled training set. Following a comparative analysis of various SSL schemes, we propose a simple SSL approach for multimodal entity disambiguation.

This framework is motivated by a naive contrastive learning idea for which we can consider the sequence multimodal data as the anchor. For example, in videos [54], a positive sample is extracted from a clip whose image corresponds to audio with the same timestamp, and the image and audio in a negative sample have different timestamps. These pseudo-labels can help to train parameters and obtain useful representations for the fine-tuning of the downstream task.

In our naive SSL framework, we manually link corresponding image-text pairs as positive samples following the

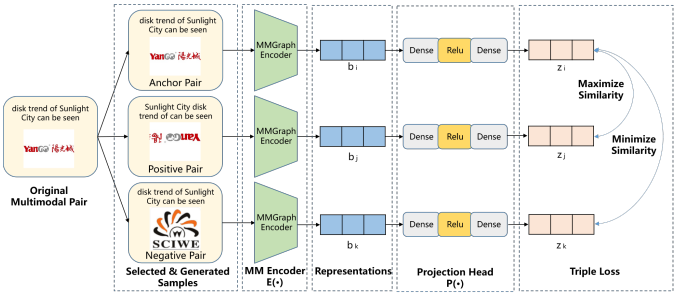


Fig. 7. SimTri. The triplet network based on contrastive and triplet loss maximizes the similarity between representations of anchor and positive samples, and minimizes the similarity between representations of anchor and negative samples.

definition and description of the entity, and disorganize the images with paired texts to form mistaken-shuffled image-text pairs as negative samples. Consequently, we can consider this SSL pretraining as a two-class classification task that determines whether an image-text pair is positive or negative. The model can learn useful representations of images and text in the pretraining task of finding the correct image-text pair from unordered data.

B. Proposed SimTri

We improve the efficiency of our self-supervised architecture and propose SSL method SimTri based on triplet loss. Siamese architectures are essential to the success of SSL methods. We modify this approach to adapt a multimodal NED task by implementing a triplet network with contrastive loss. Image comparison is accomplished by multimodal SSL on the unlabeled multimodal pairs. Because an entity mention can refer to multiple images, we choose the image (company logo) with the closest semantic distance to form anchor and positive pairs with the text, and randomly choose another image to form a negative pair. The image in the positive pair is processed with rotation and impulse noise.

The framework of SimTri in Fig. 7 has six components:

1) A batch loader module ensures that multimodal pairs in a batch have different anchor entities, which pair texts and corresponding images in the multimodal database.

2) A positive sample selection module randomly selects a multimodal pair that shares the same entity with the anchor pair. We apply simple data augmentations for image and text in positive samples, random clipping and random color distortions for positive image data, and random entity position moving in sentences for positive text data.

3) A negative sample generation module creates negative multimodal pairs. A negative multimodal pair shares the same text with the anchor sample, but has a different image corresponding to other entities.

4) A multimodal encoder $E(\cdot)$ extracts the representation vector from the anchor, positive, and negative data samples to obtain the final multimodal embeddings. For example, $b_i = E(x_i) = MMGraph(x_i)$, where $b_i \in R^d$ is the output of the final graph convolution layer.

Algorithm 1 The main learning algorithm of SimTri

Input: batch size N, constant τ, $margin$, positive and negative samples generation modules A and S, encoding and projection functions of E, P

for sampled minibatch $\{x_i\}_{i=1}^{N}$ **do**
  for all $i \in \{1, ...,\mathrm{N}\}$ **do**
    $x_{2i} = A(x_i)$ {generation of positive samples}
    $x_{3i} = S(x_i)$ {generation of negative samples}
    {multimodal encoding}
    $b_i = E(x_i)$
    $b_{2i} = E(x_{2i})$
    $b_{3i} = E(x_{3i})$
    {projection}
    $z_i = P(b_i)$
    $z_{2i} = P(b_{2i})$
    $z_{3i} = P(b_{3i})$
  end for
  for all i $\in \{1, ..., N\}$, j $\in \{N+1, ..., 2N\}$ and k $\in \{2N+1, ..., 3N\}$ **do**
    $s_{i,j} = z_i^T z_j / \|z_i\| \|z_j\|$
    $s_{i,k} = z_i^T z_k / \|z_i\| \|z_k\|$
  end for
  define $l_1(i,j) = -log(\frac{exp(s_{i,j}/\tau)}{\sum_{s=1}^{2N} \sigma_{s \neq i} exp(s_{i,s}/\tau)})$
  define $l_2(i,j,k) = max(s_{i,k} - s_{i,j} + margin, 0)$
  $L = \frac{1}{2N} \sum_{s=1}^{N} [l_1(s,2s) + l_1(2s,s)] + \sum_{s=1}^{N} l_2(s,2s,3s)$
  update networks E and P to minimize L
end for
return encoder network $E(\cdot)$

5) A multi-layer perceptron (MLP) projection head $P(\cdot)$ obtains $z_i = P(b_i) = W_2\sigma(W_1 b_i)$, where σ is ReLU nonlinearity. The MLP has one hidden layer to map the representation vector to the applied triplet contrast loss.

6) A triplet contrast loss function is proposed for the multimodal contrastive learning task. For triple group $Z = \{z_n\}$, including an anchor pair z_i, a positive pair z_j and a negative pair z_k, the triplet contrast learning task aims to learn representations such that z_i and z_j are as close as possible, and z_i and z_k are as far from each other as possible.

We randomly select a mini-batch consisting of N anchor samples. We use the positive sample selection and negative sample generation modules to expand the batch into $3N$ examples. Let $sim(x, y)$ represent the dot product (cosine similarity) between L_2 normalized x and y.

$$sim(x, y) = x^T y / \|x\| \|y\| . \tag{14}$$

For a multimodal sample $Z = \{z_n\}$, let z_i and z_j be the positive sample and negative sample, respectively, z_s a pair from other entity samples in a batch, and $2N$ the number of all anchor and positive samples. The contrastive loss function is defined as

$$L_{Z1} = -log(\frac{exp(sim(z_i, z_j)/\tau)}{\sum_{s=1}^{2N} \sigma_{s \neq i} exp(sim(z_i, z_s)/\tau)}), \tag{15}$$

where $\sigma_{k \neq i} \in \{0, 1\}$ is an index function equal to 1 if $k \neq i$ and τ is a temperature parameter. To further improve the

TABLE II
COMPARISION AND STATISTICS OF NED DATASETS ("RAW" CONNOTES UNLABELED DATA).

Statistic	NEEL	DUEL	MMFi
Train	6,025	90,000	15,718
Dev	100	10,000	1630
Test	300	10,000	1621
Raw	-	-	211,793
Total	6,425	110,000	230,762
AvgNumEntity	2.10	3.43	1.71
AvgLen	21.73	23.39	61.85

TABLE III
F1 SCORES COMPARISON OF NED RESULTS ON NEEL AND DUEL.

NEEL		DUEL	
Methods	F1	Methods	F1
FEL [61]	0.601	NTEE [71]	0.698
NTEE [71]	0.748	Fudan [72]	0.861
Mulrel-nel [73]	0.805	Mulrel-nel [73]	0.889
K-NED [5]	0.811	K-NED [5]	0.897
BERT-CRF [74]	0.831	BERT-CRF [74]	0.914
MMGraph	**0.847**	MMGraph	**0.925**

ability to discriminate positive samples with negative samples, a triplet loss function for an anchor multimodal pair Z_i is defined as

$$L_{Z2} = max(sim(z_i, z_k) - sim(z_i, z_j) + margin, 0). \quad (16)$$

The $margin$, which is set to 0.025 in this paper, prevents the distance between samples of the same category in embedding space from being too small for samples of different categories. The final loss is

$$L_Z = L_{Z1} + L_{Z2}. \quad (17)$$

As summarized in Algorithm 1, our SSL scheme is based on the architecture of a triplet network and contrastive learning to use unlabeled data for efficient training. Specifically, it finds the correlation between modalities by comparing and learning representations that can distinguish the differences between correct and unmatched multimodal pairs. The semi-supervised learning scheme based on pseudo-label self-training reinforces the self-training pipeline and can help the model to perform better at the NED task after fine-tuning the small amount of labeled data. Therefore, SimTri can conduct effective training to fit a large amount of unlabeled data, and to improve the disambiguation precision of the proposed NED methods and the entity disambiguation system.

VI. EXPERIMENT

We describe our experiments on the proposed entity disambiguation algorithm MMGraph in terms of traditional text NED, multimodal NED, and self-supervised NED.

A. Experimental Setup

1) Datasets: We experimented on Chinese and English datasets. For the Chinese disambiguation, we conducted experiments on DUEL, and evaluated methods in more detail on MMFi. We used Wikipedia as the knowledge base, and evaluated the proposed method on the widely-used NEEL dataset. To deal with English, the entity disambiguation system embeds words separated by spaces in a sentence into matrices, and puts them in a deep neural network. Unlike the Chinese NED task, the English disambiguation system does not use the word segmentation module, and the pre-trained model of BERT is based on the English version. The datasets used consisted of:

- NEEL [70]: An English NED dataset with text samples downloaded from Twitter. The training, validation, and testing sets have 6,025 tweets, 100 tweets, and 300 tweets, respectively.
- DUEL [5]: A Chinese entity disambiguation dataset including 100,000 short texts. The text corpus consists of queries and page titles. The annotated entities have instances and concepts.
- MMFi: The established MMFi includes 830 entities with corresponding multimodal visual images and S-P-O triples in the knowledge base, 18,969 annotated texts, and 211,793 unlabeled raw texts for self-supervised or weakly-supervised learning methods.

TABLE II shows the statistics and comparisons of the training, validation, and test sets.

2) Implementation Details: Our implementation was based on PyTorch with eight Nvidia RTX 2080Ti GPUs. We set the context span and length of the cropped abstract (and other S-P-O attributes) to 200, and set the input size of BERT to 400 when concatenating context span and external texts. The embedding dimensionality of text was fixed at 768, and the dimensionality was set to 768 for visual feature representations by fusing the feature vector output of the last and second-to-last stages of ResNet-50 [66]. We had two graph convolution layers separated with LeakyReLU [67]. The sizes of graph embeddings were 1024 and 768, respectively, in the first and second graph convolution layers. We used the Adam [76] optimizer with a learning rate of 10^{-7} and 100 epochs. Settings of other hyperparameters followed the same settings as NEEL [70].

We did not change hyperparameters to train the end2end system with NED methods. The maximum bidirectional matching algorithm was set to match the longest candidate string to 6, from left to right to look up the candidate mention from the dictionary. The JIEBA segmentation library and established knowledge base were combined to improve the efficiency of entity extraction.

3) Evaluation Protocols: The output results of the entity disambiguation system included specific named entities in the text and their disambiguation results. The NED evaluation calculates Precision, Recall, and F1 score (F1) to show the performance of models by comparing model output and manual labeling results for the extracted entity mentions. The performance of a model mainly depends on the final

TABLE IV
COMPARISON OF NED RESULTS ON MMFI IN SINGLE TEXT MODALITY (ST) AND MULTIMODAL (MM) TASKS.

Methods	NTEE [71]	FUDAN [72]	Mulrel-nel [73]	K-NED [5]	INEDGE [22]	BERT-CRF [74]	DCA-SL+Triples [75]	MMGraph-ST	MMGraph-MM
Precision	0.7909	0.8313	0.8371	0.8704	0.8892	0.9076	0.9155	0.9392	**0.9431**
Recall	0.8065	0.8374	0.8467	0.9013	0.9065	0.9325	0.9412	0.9484	**0.9576**
F1	0.7986	0.8343	0.8419	0.8856	0.8978	0.9198	0.9281	0.9438	**0.9503**

F1 score. Given a text input T, the ground truth of N mention words in T includes entity mentions m, positions p, and linking entity ids e. Let grount truth be $GR_T = \{(m_1, p_1, e_1), ..., (m_k, p_k, e_k)\}$. Accordingly, the output result of the model is $GR'_T = \{(m_1, p_1, e_1), ..., (m_n, p_n, e_n)\}$.

The evaluation metrics were

$$Precision = \frac{\sum_{t \in T} |GR_T \bigcap GR'_T|}{GR'_T}, \tag{18}$$

$$Recall = \frac{\sum_{t \in T} |GR_T \bigcap GR'_T|}{GR_T}, \tag{19}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}. \tag{20}$$

4) Baseline Methods: We compared the effectiveness of our model with that of eight state-of-the-art methods in the case of short text. These were:

- FEL [61]: A tool for training models and linking entities to knowledge bases in documents and queries. The Deep Structured Semantic Model (DSSM) [77] is used for comparing the entity embedding.
- NTEE [71]: A neural network model that learns the embeddings of text and Wikipedia entities, and uses them for entity disambiguation.
- Fudan [72]: A system includes entity disambiguation and entity linking modules with a Chinese NED service API provided by Fudan University.
- Mulrel-nel [73]: An advanced neural implementation method of multi-relational NED.
- BERT-CRF [74]: A baseline BERT method using similarity of context and external knowledge to extract semantic vectors.
- K-NED [5]: A knowledge-enhanced approach to short-text entity disambiguation also proposed in [5].
- INEDGE [22]: A latest graph-based NED method that can use the global prior knowledge in data.
- DCA-SL+Triples [75]: A state-of-the-art NED method based on BERT and knowledge graph triples.

We experimented with default hyperparameters for these methods on three datasets.

B. Evaluation

1) Comparison with Existing Models: TABLE III compares the performance on the NEEL and DUEL datasets in traditional supervised textual NED tasks and shows that our method outperformed eight state-of-the-art models. Higher F1 values indicate the superiority of the proposed MMGraph. FEL, NTEE, Fudan, and Mulrel-nel are shallow neural networks, with limited performance on large-scale datasets. Compared with BERT-CRF, MMGraph, which adopts advanced graph convolution, can easily capture the context in natural languages. Although K-NED also uses the graph structure, the joint model of BERT and graph convolution enables MMGraph to obtain deeper and more precise semantic features for further disambiguation. The F1 performance of MMGraph was 1.6% and 1.1% higher than the state-of-the-art model BERT-CRF on NEEL and DUEL, respectively. This is because word segmentation in Chinese text affects the original context, making it harder for a GCN to capture the semantic relationship among contexts. A manually fragmentized sentence loses more context than segmented words in English. However, graph convolution can still extract useful correlations in segmented sentences. Our approach disambiguates all the correct results from the examples in Fig. 3. This shows that the graph structure and corresponding graph representations can improve the performance of short-text NED.

We also compared our methods with three state-of-the-art approaches on MMFi. TABLE IV shows the test results, and the results of our models with and without multimodal data, where MMGraph-ST presents the text baseline of the model without visual information, and MMGraph-MM performs entity disambiguation with the help of multimodal data. We also reproduced the experiments of INEDGE [22] and DCA-SL+Triples [75] on MMFi since it has the external knowledge in the form of S-P-O triples for implementing prior graph structure. We can see that the F1-scores of the INEDGE method based on the DeepWalk shallow graph model were 2.2% and 4.6% lower than those of BERT-CRF and MMGraph-ST, respectively, which shows that MMGraph was superior to the compared models, including the pretrained contextual language model BERT-CRF. Compared with the state-of-the-art NED method DCA-SL+Triples, the F1-score of MMGraph-ST is also higher by 1.57%. It shows that the designed prior graph structure in MMGraph-ST is more suitable for this NED task. We can see from the table that F1 increased by 0.65% from 94.38% to 95.03% when we used multimodal graph convolution to obtain the fusion of visual and textual features, because MMGraph-MM can find the latent semantic correlation in textual sentences and visual logos bridging the gap between text and images. In theory, the use of visual logos can help differentiate company entities and texts that are naturally similar. This can be regarded as a pseudo-label that expands the model capacity to improve performance, and therefore avoids overfitting and brings about robust generalization [78]. The experiments showed that the proposed graph embedding on texts is already better than

TABLE V
RESULTS OF ABLATION STUDIES FOR PROPOSED MMGRAPH ON MMFI.

Methods	Precision	Recall	F1
Word2Vec	0.5987	0.6131	0.6058
BERT-CRF	0.8704	0.9013	0.8856
BERT-CRF+CNN	0.8736	0.9138	0.8932
BERT-CRF+GCN	0.9381	0.9473	0.9427
MMGraph	**0.9431**	**0.9576**	**0.9503**

previous methods, and multimodal graph convolution is even more beneficial in the NED task.

2) Ablation Studies: To more deeply understand our model, ablation studies were performed to analyze the effect of the proposed multimodal graph embeddings. TABLE V shows the results of ablation studies, which demonstrate the crucial role of the proposed multimodal graph convolution in improving NED performance. First, by calculating the similarity between the word vector of context and the candidate entity description in the knowledge base, a naive baseline method based on Word2Vec [23] was implemented on MMFi. We also observed the performance of BERT-CRF as a deep method baseline. Because the proposed method was based on the comparison of context and knowledge base external information, we used CNN to replace the GCN in MMGraph and also to extract context representation in ablation studies.

We compared the performance after adding different parts of context embedding. In TABLE V, BERT-CRF+CNN indicates the addition of the two-dimensional convolution layer to BERT-CRF to capture the correlation between the mention context and external texts, and BERT-CRF+GCN replaces traditional convolution with the proposed graph convolution. MMGraph refers to the result of our complete model. The F1-score of BERT-CRF was 88.56%, which is outstanding when compared with previous models. The addition of a two-dimensional convolution module increased the F1-score by 0.67%. Compared with BERT-CRF, BERT-CRF+GCN increased the F1-score from 88.56% to 94.27%. We find a 4.95% improvement in the proposed graph convolution compared with traditional convolution, and a further 0.76% improvement in multimodal graph embedding on MMFi. This demonstrates the effectiveness of using visual features, because multimodal graph convolution can take advantage of the interaction between word embeddings in sentences and visual features in images to improve the features of final embeddings.

3) Effect of Self-Supervised Learning: To explore the effects of the proposed SSL methods, we present the performance of different learning strategies on MMFi in Fig. 8. In the experiments, the models were pretrained in a task-agnostic way, following the SSL schemes to learn multimodal representations. The pretrained models were fine-tuned and tested on the downstream NED task. Comparing the results in Fig. 8, we see that SSL on unlabeled data had a better effect (96.20% F1-score) than the supervised methods (95.03% F1-score), without the need for rich semantic labels.

A self-training method based on pseudo-labels was selected as the baseline method in the SSL task, and was compared with

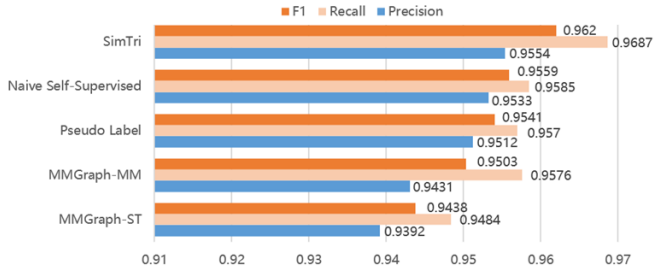


Fig. 8. Comparison of entity disambiguation performance for self-supervised learning methods on MMFi. MMGraph-ST: MMGraph using only text data; MMGraph-MM: MMGraph using multimodal data.

Fig. 9. Error analysis of our proposed MMGraph model on the MMFi. Two failure cases are demonstrated, where the images are shown on the left, and the sentences (translated from the Chinese) with ground truth and prediction are shown on the right. The label of 0 indicates the negative entities that are not included in the knowledge base, while the label of 0 indicates the positive entities that are included in the knowledge base. The red color refers to the wrong NED results.

supervised results. It was seen that self-supervised pretraining could produce positive effects for a model to learn a better representation, and the pretrained model could better perform entity disambiguation after fine-tuning on a small amount of labeled data. Naive and novel self-supervised approaches were evaluated on unlabeled multimodal data using data augmentation and the proposed SimTri. In Fig. 8, a higher F1-value compared with other schemes shows the superiority of SimTri.

C. Discussion

The superior performance of the proposed MMGraph is based on prior knowledge of the graph structure. TABLE III and TABLE IV indicate that a prior graph structure and corresponding graph representations encode some complementary information. The graph structure used in this study takes advantage of prior knowledge, such as the sentence structure, which is not relevant to a specific entity, to improve the ability of disambiguation. For example, we observe that an ambiguous entity mention tends to be at the rear of a sentence and is less likely to be influenced by the beginning, such as the subject. Furthermore, the transmission of information in non-Euclidean space greatly improves model data fitting and the efficiency of graph embeddings. The proposed graph multimodal convolution utilizes the correlation matrix based on data analysis, grammatical structure, and multimodal data

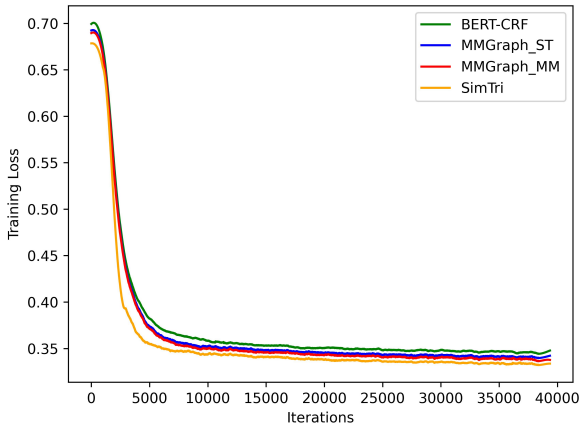


Fig. 10. Training loss curves with same number of iterations on BERT-CRF, MMGraph-ST, MMGraph-MM, and SimTri.

of the external knowledge, and learns a stronger representation for short text, so as to enhance the disambiguation ability of the model and improve accuracy.

In designing the graph structure, we used the NLPIR word segmentation system to analyze ambiguous sentence structures and parts of speech in context. For example, the position, offset, and grammar rules of various parts in sentences such as the four words before and after the entities were recorded by NLPIR. Through the analysis of prior probability, we found that ambiguous sentences have a certain regularity. For example, the correlation effect of entity type on the rest of the words is related to the positional structure of the sentence. The prior knowledge described above helped in the design of our graph structure. Thus, the proposed graph embedding can utilize the designed structure and make use of prior information to help the model judge whether an entity mention in a specific context is ambiguous.

The proposed SimTri implements SSL via contrast loss and triplet loss. Based on graph regularization theory, it utilizes the structure of a multimodal graph to obtain better embeddings. Furthermore, we used aligned multimodal information in unlabeled data as pseudo-labels to help the model memorize which image a real text corresponds to, while improving the ability to learn representations for SSL and ensuring better performance in downstream NED. Specifically, with the intervention of triplet loss, the model learns multimodal representations that can distinct positive samples from negative samples during training, and also learns the representative features that make the original positive samples and the generated hand-crafted pseudo-positive samples have the same pattern. Therefore, features that the model learns can be more abundant and useful for downstream entity disambiguation. Through advanced model design and SSL enhancement, the disambiguation system implemented in this paper can realize a superior disambiguation function with relatively high accuracy.

To better understand the proposed methods, and in particular to explore the limitations of MMGraph with SimTri, we analyzed the failure cases on MMFi in Fig. 9. After the pretraining of SimTri, the text has been well-aligned with the visual information. For the first case, MMGraph fails to disambiguate the mention ”Lujiazui” due to the confusing context. The mention ”Lujiazui” can both refers to a location entity or a corporation entity. Even with the help of multimodal information and S-P-O triples from the external knowledge base (KB), the context in the original text is still disturbing and brings bias to the model. Specifically, ”Lujiazui” occurs at the end of the sentence, which is uncommon in the positive samples of corporation entities, and is influenced by the first word Shanghai. These are all the factors that can make MMGraph wrongly regards ”Lujiazui” as a location entity during inference. The second case indicates another challenge. The mention ”Green Power” can refer to several cooperation entities (including Beijing Green Power Co. LTD), and the visual logos of this corporation can be similar in the green color. Thus, MMGraph wrongly predicts ”Green Power” as a corporation entity in KB with an index of 324. However, the ”Green Power” in the context is an Indonesian company that is not listed on the A-share market, so it is not included in our knowledge base.

In addition, we conducted the convergence analysis on the proposed algorithms and also on the baseline BERT-CRF to better explore the effect of the proposed multimodal graph convolution and the proposed SSL method SimTri. The loss during the training is shown in Fig. 10 with respect to iterations. As shown in Fig. 10, faster convergence is observed for MMGraph-ST and MMGraph-MM compared to BERT-CRF. We notice that the multimodal version of MMGraph obtains a lower training loss in the final iterations, which means the proposed multimodal convolution improves the generalization performance of the model and reduces the model error. The training loss based on the pretrained model via SimTri firstly decreases faster than the supervised models, then converges smoothly and acquires the lower convergence loss than the other three models. It illustrates that the proper SSL pretraining approach benefits the convergence of supervised training via learning the representation on multimodal datasets in the contrastive task-agnostic pretraining.

VII. Conclusion

We analyzed the features of sentence structure and characteristics of multimodal data, and proposed the MMGraph approach for multimodal NED. An effective entity disambiguation pipeline was obtained by embedding MMGraph in an end-to-end entity disambiguation system. Furthermore, the proposed self-supervised learning framework SimTri cam learn better representations from multimodal unlabeled data. We established the MMFi database to evaluate these methods. Experiments on MMFi and two widely used NED datasets showed that MMGraph performed best in supervised NED tasks, and that SimTri could further improve F1-values, which reflects the potential of SSL for NED tasks.

We hope that the supervised MMGraph and self-supervised SimTri can benefit the intelligent information processing in NLP community and help the multimedia community, such as multimodal knowledge graph research. In future work, we

hope to combine newer graph learning methods such as the adaptive graph learning model [79] with the proposed framework to further improve entity disambiguation performance.

REFERENCES

[1] W. Shen, J. Wang, and J. Han, "Entity linking with a knowledge base: Issues, techniques, and solutions," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 2, pp. 443–460, 2014.

[2] P. Radhakrishnan, P. Talukdar, and V. Varma, "Elden: Improved entity linking using densified knowledge graphs," in *Proc. NAACL-HLT*, 2018.

[3] O.-E. Ganea and T. Hofmann, "Deep joint entity disambiguation with local neural attention," in *Proc. EMNLP*, 2017.

[4] H. Shahbazi, X. Z. Fern, R. Ghaeini *et al.*, "Joint neural entity disambiguation with output space search," in *Proc. Int. Conf. Comput. Linguist.*, 2018.

[5] Z. Feng, Q. Wang, W. Jiang, Y. Lyu, and Y. Zhu, "Knowledge-enhanced named entity disambiguation for short text," in *Proc. AACL/IJCNLP*, 2020.

[6] F. Yao, X. Sun, H. Yu, W. Zhang, W. Liang, and K. Fu, "Mimicking the brain's cognition of sarcasm from multidisciplines for twitter sarcasm detection," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–15, 2021.

[7] Y. Liu, H. Li, A. Garcia-Duran *et al.*, "Mmkg: multi-modal knowledge graphs," in *Proc. Europ. Semant. Web Conf.*, 2019.

[8] S. Ji, S. Pan, E. Cambria, P. Marttinen, and P. S. Yu, "A survey on knowledge graphs: Representation, acquisition, and applications," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–21, 2021.

[9] A. V. Kannan, D. Fradkin, I. Akrotirianakis *et al.*, "Multimodal knowledge graph for deep learning papers and code," in *Proc. ACM Int. Conf. Inf. Knowl. Manag.*, 2020.

[10] C. Y. Li, X. Liang, Z. Hu, and E. P. Xing, "Knowledge-driven encode, retrieve, paraphrase for medical image report generation," in *Proc. AAAI Conf. Artif. Intell.*, 2019.

[11] S. Moon, L. Neves, and V. Carvalho, "Zeroshot multimodal named entity disambiguation for noisy social media posts," in *Proc. ACL*, 2018.

[12] X. Liu, F. Zhang, Z. Hou *et al.*, "Self-supervised learning: Generative or contrastive," *arXiv preprint arXiv:2006.08218*, 2020.

[13] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon *et al.*, "Learning deep representations by mutual information estimation and maximization," in *Proc. Int. Conf. Learn. Representations*, 2019.

[14] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020.

[15] L. Floridi and M. Chiriatti, "Gpt-3: Its nature, scope, limits, and consequences," *Minds Mach.*, vol. 30, no. 4, pp. 681–694, 2020.

[16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, 2019.

[17] T. N. Kipf and W. Max, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Representations*, 2017.

[18] R. Navigli, "Word sense disambiguation: A survey," *ACM Comput. Surv.*, vol. 41, no. 2, pp. 10:1–10:69, 2009.

[19] M. Strube and S. P. Ponzetto, "Wikirelate! computing semantic relatedness using wikipedia," in *Proc. AAAI Conf. Artif. Intell.*, 2006.

[20] D. Milne and I. H. Witten, "Learning to link with wikipedia," in *Proc. ACM Int. Conf. Inf. Knowl. Manag.*, 2008.

[21] N. Kolitsas, O.-E. Ganea, and T. Hofmann, "End-to-end neural entity linking," in *Proc. Conf. Comput. Nat. Lang. Learn.*, 2018.

[22] Ö. Sevgili, A. Panchenko, and C. Biemann, "Improving neural entity disambiguation with graph embeddings," in *Proc. ACL*, 2019.

[23] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. Int. Conf. Learn. Representations*, 2013.

[24] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proc. Int. Conf. Mach. Learn.*, 2014.

[25] M. Francis-Landau, G. Durrett, and D. Klein, "Capturing semantic similarity for entity linking with convolutional neural networks," in *Proc. NAACL-HLT*, 2016.

[26] S. Broscheit, "Investigating entity knowledge in BERT with simple neural end-to-end entity linking," in *Proc. Conf. Comput. Nat. Lang. Learn.*, 2019.

[27] H. Huang, L. Heck, and H. Ji, "Leveraging deep neural networks and knowledge graphs for entity disambiguation," *arXiv preprint arXiv:1504.07678*, 2015.

[28] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, 2014.

[29] O. Adjali, R. Besançon, O. Ferret, H. Le Borgne, and B. Grau, "Multimodal entity linking for tweets," in *Proc. Europ. Conf. Inf. Retr.*, 2020.

[30] M. Gori, G. Monfardini, and F. Scarselli, "A new model for learning in graph domains," in *Proc. IEEE Int. Jt. Conf. Neural Netw.*, 2005.

[31] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61–80, 2009.

[32] P. Velickovic, G. Cucurull, A. Casanova *et al.*, "Graph attention networks." in *Proc. Int. Conf. Learn. Representations*, 2018.

[33] W. L. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017.

[34] D. Yuan, X. Chang, P.-Y. Huang, Q. Liu, and Z. He, "Self-supervised deep correlation tracking," *IEEE Trans. Image Process.*, vol. 30, pp. 976–985, 2020.

[35] E. Strubell, P. Verga, D. Andor, D. Weiss, and A. McCallum, "Linguistically-informed self-attention for semantic role labeling," in *Proc. Conf. EMNLP*, 2018.

[36] B. Li, X. Shu, and R. Yan, "Storyboard relational model for group activity recognition," in *Proc. ACM Int. Conf. Multimedia Asia*, 2021, pp. 1–7.

[37] C. Bai, A. Zheng, Y. Huang, X. Pan, and N. Chen, "Boosting convolutional image captioning with semantic content and visual relationship," *Displays*, vol. 70, p. 102069, 2021.

[38] P. Zhou, C. Bai, J. Xia, and S. Chen, "Cmrdf: A real-time food alerting system based on multimodal data," *IEEE Internet Things J.*, pp. 1–1, 2020.

[39] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2020.

[40] R. Zhang, P. Isola, and A. A. Efros, "Split-brain autoencoders: Unsupervised learning by cross-channel prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2017.

[41] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *Proc. Eur. Conf. Comput. Vis.*, 2016.

[42] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *Proc. Eur. Conf. Comput. Vis.*, 2016.

[43] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2016.

[44] C. Doersch and A. Zisserman, "Multi-task self-supervised visual learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017.

[45] A. Dosovitskiy, L. Beyer, A. Kolesnikov *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Representations*, 2021.

[46] X. Chen, H. Fan, R. B. Girshick, and K. He, "Improved baselines with momentum contrastive learning," *arXiv preprint arXiv:2003.04297*, 2020.

[47] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a siamese time delay neural network," in *Proc. Adv. Neural Inf. Process. Syst.*, 1993.

[48] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2006.

[49] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *Proc. Int. Conf. Artif. Intell. Stat.*, 2010.

[50] K. He, H. Fan, Y. Wu, S. Xie, and R. B. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2020.

[51] J.-B. Grill, F. Strub, F. Altché *et al.*, "Bootstrap your own latent: A new approach to self-supervised learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020.

[52] X. Chen and K. He, "Exploring simple siamese representation learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2021.

[53] M. Patrick, Y. M. Asano, R. Fong *et al.*, "Multi-modal self-supervision from generalized data transformations," *arXiv preprint arXiv:2003.04298*, 2020.

[54] A. Owens and A. A. Efros, "Audio-visual scene analysis with self-supervised multisensory features," in *Proc. Eur. Conf. Comput. Vis.*, 2018.

[55] B. Korbar, D. Tran, and L. Torresani, "Cooperative learning of audio and video models from self-supervised synchronization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018.

[56] I. Misra, C. L. Zitnick, and M. Hebert, "Shuffle and learn: Unsupervised learning using temporal order verification," in *Proc. Eur. Conf. Comput. Vis.*, 2016.

[57] D. Wei, J. Lim, A. Zisserman, and W. T. Freeman, "Learning and using the arrow of time," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2018.

[58] D. Xu, J. Xiao, Z. Zhao *et al.*, "Self-supervised spatiotemporal learning via video clip order prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2019.

[59] L. Zhen, P. Hu, X. Peng, R. S. M. Goh, and J. T. Zhou, "Deep multimodal transfer learning for cross-modal retrieval," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–13, 2020.

[60] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. E. Hinton, "Big self-supervised models are strong semi-supervised learners," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020.

[61] R. Blanco, G. Ottaviano, and E. Meij, "Fast and space-efficient entity linking for queries," in *Proc. ACM Int. Conf. Web Search Data Min.*, 2015.

[62] W. Che, Y. Feng, L. Qin, and T. Liu, "N-ltp: A open-source neural chinese language technology platform with pretrained models," *arXiv preprint arXiv:2009.11616*, 2020.

[63] L. Zhou and D. Zhang, "Nlpir: A theoretical framework for applying natural language processing to information retrieval," *J. Am. Soc. Inf. Sci. Technol.*, vol. 54, no. 2, pp. 115–123, 2003.

[64] M. Sun, X. Chen, K. Zhang, Z. Guo, and Z. Liu, "Thulac: An efficient lexical analyzer for chinese," 2016.

[65] A. Vaswani, N. Shazeer, N. Parmar *et al.*, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017.

[66] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016.

[67] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. Int. Conf. Mach. Learn.*, 2013.

[68] N. Rasiwasia, J. C. Pereira, E. Coviello *et al.*, "A new approach to cross-modal multimedia retrieval," in *Poc. ACM Int. Conf. Multimedia*, 2010.

[69] D. R. Hardoon, S. R. Szedmak, and J. R. Shawe-taylor, "Canonical correlation analysis: An overview with application to learning methods," *Neural Comput.*, vol. 16, no. 12, pp. 2639–2664, 2004.

[70] G. Rizzo, B. Pereira, A. Varga, M. Van Erp, and A. E. Cano Basave, "Lessons learnt from the named entity recognition and linking (neel) challenge series," *Semant. Web*, vol. 8, no. 5, pp. 667–700, 2017.

[71] I. Yamada, H. Shindo, H. Takeda, and Y. Takefuji, "Learning distributed representations of texts and entities from knowledge base," *Trans. Assoc. Comput. Linguist.*, vol. 5, pp. 397–411, 2017.

[72] B. Xu, Y. Xu, J. Liang *et al.*, "Cn-dbpedia: A never-ending chinese knowledge extraction system," in *Proc. IEA/AIE*, 2017.

[73] P. Le and I. Titov, "Improving entity linking by modeling latent relations between mentions," in *Proc. ACL*, 2018.

[74] J. Cheng, C. Pan, J. Dang *et al.*, "Entity linking for chinese short texts based on bert and entity name embeddings," in *Proc. China Conf. Knowl. Graph Semant. Comput.*, 2019.

[75] I. O. Mulang', K. Singh, C. Prabhu, A. Nadgeri, J. Hoffart, and J. Lehmann, "Evaluating the impact of knowledge graph context on entity disambiguation models," in *Proc. ACM Int. Conf. Inf. Knowl. Manag.*, 2020.

[76] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015.

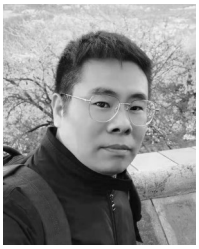
[77] P.-S. Huang, X. He, J. Gao, L. Deng, A. Acero, and L. Heck, "Learning deep structured semantic models for web search using clickthrough data," in *Proc. ACM Int. Conf. Inf. Knowl. Manag.*, 2013.

[78] Y. Huang, C. Du, Z. Xue, X. Chen, H. Zhao, and L. Huang, "What makes multimodal learning better than single (provably)," *arXiv preprint arXiv:2106.04538*, 2021.

[79] R. Zhou, X. Chang, L. Shi, Y.-D. Shen, Y. Yang, and F. Nie, "Person reidentification via multi-feature fusion with adaptive graph learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 5, pp. 1592–1601, 2019.

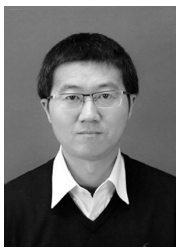
Pengfei Zhou received the B.E. degree from Zhejiang University of Technology, Hangzhou, China, in 2021. He is currently pursuing the M.E. degree in the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. His researching interests include multimedia processing and computer vision.

Kaining Ying received the B.E. degree from Zhejiang University of Technology, Hangzhou, China, in 2021. He is currently pursuing the M.E. degree in the College of Computer Science, Zhejiang University of Technology, supervised by Dr. Zhenhua Wang. His research interests include deep learning and multimedia processing.



Zhenhua Wang received the B.E. degree in 2007, and the M.E. degree in 2010, both from Northwest A&F University, China. He received the Ph.D. degree in computer vision from The University of Adelaide, Adelaide, SA, Australia, in 2014. He is a lecturer in the College of Computer Science, Zhejiang University of Technology, Hangzhou, China. His research interests include computer vision and statistical machine learning.

Dongyan Guo received the B.S. degree in application mathematics and the Ph.D. degree in pattern recognition and intelligent systems from the Nanjing University of Science and Technology, China, in 2008 and 2015, respectively. Since 2015, he has been an Associate Profess with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, China. His research interests include computer vision and machine learning.



Cong Bai received the B.E. degree from Shandong University, Jinan, China, in 2003, the M.E. degree from Shanghai University, Shanghai, China, in 2009, and the Ph.D. degree from the National Institute of Applied Sciences, Rennes, France, in 2013. He is an Associate Professor and Ph.D supervisor with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, China. His research interests include computer vision and multimedia processing.