Integrating Manifold Knowledge for Global Entity Linking with Heterogeneous Graphs Postprint

Authors: Zhibin Chen, Yuting, Wu, Yansong, Feng, Dongyan Zhao, Yuting, Wu

Date: 2022-11-28T00:00:00+00:00

Abstract

Entity Linking (EL) aims to automatically link the mentions in unstructured documents to corresponding entities in a knowledge base (KB), which has recently been dominated by global models. Although many global EL methods attempt to model the topical coherence among all linked entities, most of them failed in exploiting the correlations among manifold knowledge helpful for linking, such as the semantics of mentions and their candidates, the neighborhood information of candidate entities in KB and the fine-grained type information of entities. As we will show in the paper, interactions among these types of information are very useful for better characterizing the topic features of entities and more accurately estimating the topical coherence among all the referred entities within the same document. In this paper, we present a novel HEterogeneous Graph-based Entity Linker (HEGEL) for global entity linking, which builds an informative heterogeneous graph for every document to collect various linking clues. Then HEGEL utilizes a novel heterogeneous graph neural network (HGNN) to integrate the different types of manifold information and model the interactions among them. Experiments on the standard benchmark datasets demonstrate that HEGEL can well capture the global coherence and outperforms the prior state-of-the-art EL methods.

Full Text

Preamble

RESEARCH PAPER ChinaXiv 合作期刊
Integrating Manifold Knowledge for Global Entity Linking with Heterogeneous Graphs
Zhibin Chen[1,2], Yuting Wu[1,3]†, Yansong Feng[1,3] & Dongyan Zhao[1,3]

[1]Wangxuan Institute of Computer Technology, Peking University, Beijing 100871, China
[2]Center for Data Science, Peking University, Beijing 100871, China
[3]The MOE Key Laboratory of Computational Linguistics, Peking University, Beijing 100871, China

Keywords: Entity linking; Heterogeneous graph; Graph neural network; Entity disambiguation; Knowledge base

Citation: Chen, Z.B., et al.: Integrating manifold knowledge for global entity linking with heterogeneous graphs. Data Intelligence 4(1), 20-40 (2022). doi: 10.1162/dint_a_{00116}
Received: October 30, 2021; **Revised:** December 21, 2021; **Accepted:** January 11, 2022

Corresponding author: Yuting Wu (Email: wyting@pku.edu.cn; ORCID: 0000-0002-7550-3804).

ABSTRACT

Entity Linking (EL) aims to automatically link mentions in unstructured documents to corresponding entities in a knowledge base (KB), a task recently dominated by global models. Although many global EL methods attempt to model topical coherence among all linked entities, most fail to exploit correlations among manifold knowledge helpful for linking, such as the semantics of mentions and their candidates, neighborhood information of candidate entities in the KB, and fine-grained type information of entities. As we demonstrate in this paper, interactions among these information types are highly useful for better characterizing the topic features of entities and more accurately estimating topical coherence among all referred entities within the same document. We present a novel **HEterogeneous Graph-based Entity Linker (HEGEL)** for global entity linking, which builds an informative heterogeneous graph for every document to collect various linking clues. HEGEL then utilizes a novel heterogeneous graph neural network (HGNN) to integrate different types of manifold information and model interactions among them. Experiments on standard benchmark datasets demonstrate that HEGEL effectively captures global coherence and outperforms prior state-of-the-art EL methods.

1. INTRODUCTION

Entity Linking (EL) is the task of mapping entity mentions with specified context in an unstructured document to corresponding entities in a given Knowledge Base (KB). This bridges the gap between abundant unstructured text in large corpora and structured knowledge sources, thereby supporting many

knowledge-driven natural language processing (NLP) tasks and methods, such as question answering [**?**], text classification [**?**], information extraction [**?**], and knowledge graph construction [**?**].

Recently, the EL task has been dominated by global methods [**?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**, **?**], which model topical coherence among linked entities of mentions in the same document. Global information relies on the semantic and topical coherence of entities related to various mentions within the same document, which most state-of-the-art models integrate with local mention-contextual information to alleviate biases from local contextual information. For instance, as shown in Figure 1 [Figure 1: see original paper], when linking the mention "England," it is difficult to decide between the candidate entities *England national football team* and *England national rugby union team* using only the surrounding sports-related local context containing match scores or stadium names, which may contain noise and lead to linking to the more popular but incorrect candidate *England national football team*. However, if an EL model can capture the topical coherence of the common topic "rugby" among all mentions "Scotland," "Murrayfield," "Cuttitta," and "England" in the current paragraph—such as by considering the nearby mention "Cuttitta," which links to the candidate *Marcello Cuttitta*, a former Italian rugby union player—the model can correctly link the mention "England" to the candidate *England national rugby union team*.

Although prior global EL approaches have greatly boosted performance over local models, most do not simultaneously consider multiple types of useful information and their interactions—such as the semantics of mentions and their candidates, neighborhood information of candidate entities in the KB, and fine-grained type information of entities—when modeling global coherence, and thus fail to precisely estimate coherence among referred entities. As we demonstrate in this paper, effectively modeling interactions among manifold linking knowledge can help better model topical coherence and achieve more accurate EL.

Most recently, some global methods [**?**, **?**] construct a document-level graph with candidate entities of mentions as nodes and exploit Graph Convolutional Networks (GCN) [**?**] on the graph to integrate global information, delivering promising results. Inspired by the effectiveness of using GCN to model global signals, we present **HEterogeneous Graph-based Entity Linker (HEGEL)**, a novel global EL framework designed to model interactions among manifold heterogeneous information from different sources by constructing a document-level informative heterogeneous graph and applying a heterogeneous architecture in GNN aggregation operations. We first construct a document-level informative heterogeneous graph with mentions, candidate entities, neighbors of entities, and extracted keywords as nodes, creating different types of edges to link these different node types. We then apply a meticulously designed heterogeneous graph neural network (HGNN) on the constructed heterogeneous graph to encode global coherence, which allows information propagation along the informative graph structure and encourages sufficient interactions among different information types. Followed by traditional scoring combination and ranking

procedures, our model can be trained end-to-end.

Our contributions can be summarized as follows:

- We designed a novel approach to construct a document-level informative heterogeneous graph that collects manifold linking knowledge from different sources to support the linking process.
- We proposed a meticulously designed heterogeneous graph neural network on the constructed graph, which integrates different sources of information and encourages sufficient interactions among them, more precisely characterizing the topic features of candidate entities and better capturing topical coherence. To the best of our knowledge, this is the first work to employ a heterogeneous graph neural network in Entity Linking tasks.
- Extensive experiments and analysis on six standard EL datasets demonstrate that our HEGEL achieves state-of-the-art performance over mainstream EL methods.

2.1 Entity Linking

Most existing models use not only local methods relying on local context of individual mentions independently [**?**, **?**, **?**, **?**, **?**, **?**], but also global methods considering coherence among linked entities of all mentions by jointly linking across the entire document [**?**, **?**]. Local methods typically make use of extracted local features through feature engineering, including pairwise statistical features like Wikipedia linking frequency and similarity scores between mentions and candidate entities, such as mention-entity similarities implemented as cosine similarities between document local contexts and entity Wikipedia titles [**?**]. Recently, Pretrained Language Models (PLMs), which achieve leading performance in other NLP tasks, have also been used in local linking models. PLM-based linking models focus on unique settings such as zero-shot [**?**] and multilingual [**?**] scenarios to exploit PLM superiority in understanding tasks under these settings. To alleviate noise from local information, global methods model semantic coherence and relationships between linked entities within the same document. As global coherence optimization is NP-hard, different approximation methods are often employed. Apart from traditional methods like loopy belief propagation [**?**, **?**], several works approximate the problem as a sequential decision problem [**?**] or graph learning [**?**, **?**, **?**, **?**].

Following graph-based neural network modeling methods, HEGEL expands graph utilization in the EL task to a heterogeneous style, which not only enjoys the strong representation ability of heterogeneous graph structure but also becomes effective by avoiding additional inference steps required in sequence-style models.

2.2 Graph Neural Networks

Graph Neural Network (GNN) is a strong and flexible framework for learning on data with graph structure. Since the emergence of Graph Convolutional Network (GCN) [**?**], GNN has been increasingly widely used in many tasks, with several popular architectures such as GraphSAGE [**?**] and GAT [**?**] proposed to learn representations on graphs. The natural graph structure entailed in EL tasks provides a favorable condition for applying GNN methods to model global information. NCEL [**?**] performs GCN on constructed subgraphs for every mention, where nodes are entity candidates of the current mention and surrounding mentions with edges linking from the former to the latter. SGEL [**?**] combines features of mention-by-mention sequential models and GAT by building a graph containing previously predicted entities, current candidates, and later unpredicated mention candidates as nodes. GNED [**?**] builds a homogeneous graph by embedding entities and words into the same vector space and extracts words from descriptions and context in the KB for every candidate to form nodes and edges.

With the emergence of massive heterogeneous information, many heterogeneous GNN works have proven effective. The mainstream of HGNN models is based on metapath construction [**?**], but several HGNN architectures free of metapath have been proposed recently [**?**]. Our HEGEL follows these works and utilizes heterogeneous structure to model interactions among different types of linking information.

---

3. PROBLEM FORMULATION

Given a list of entity mentions $M = \{m_1, \ldots, m_{|M|}\}$ in a document D, the EL task can be formulated as linking each mention m_i to its corresponding entity e_i^* from the entity collection E of KB or NIL (i.e., $e_i^* = \text{NIL}$, meaning mention m_i cannot be reasonably linked to any corresponding entity in E). Generally speaking, EL methods consist of two stages.

3.1 Candidate Generation Stage

EL tasks typically begin by generating a small list of candidate entities C_i for mention m_i due to the unacceptable computational cost of traversing the entire entity collection E. For candidate generation, we use the method proposed in [**?**, **?**], which employs (1) computed mention-entity prior $\widehat{p}(e|m)$ and (2) local context-entity similarity, calculated as the similarity between candidate entity embeddings and average embeddings of context words.

This stage aims to contain the correct entity e_i^*, and the ratio of candidate lists containing the corresponding entity is referred to as the recall of candidate generation.

3.2 Candidate Disambiguation Stage

In this stage, EL methods assign a score calculated by the EL model to each candidate $e \in C_i$ and select the top-ranked candidate as the predicted answer, or predict NIL under specified situations. Most EL methods, including this work, focus on improving performance at this stage. As mentioned in Section 2.1, different local and global models calculate linking scores. Local methods focus on the corresponding mention itself, regardless of other mentions or linked entities. That is, local methods address the linking problem through independent calculation for every mention:

$$\arg\max_{e \in C_i} \Psi_{\text{Local}}(m_i, e)$$

where Ψ_{Local} is a scoring function for the mention-entity pair. Unlike local methods without inter-mention interaction, interdependencies measured by global methods can be generally represented as a coherence scoring function that takes into account entity topic coherence:

$$\arg \max_{e_1,\ldots,e_{|M|} \in C_1 \times \cdots \times C_{|M|}} \sum_{i=1}^{|M|} \Psi_{\text{Local}}(m_i, e_i) + \Phi_{\text{Global}}(E, D)$$

where $E = \{e_1, \ldots, e_{|M|}\}$ is the predicted entity list for entity mentions M of document D, and $\Phi_{\text{Global}}(E, D)$ is the global function measuring how entities cohere with each other.

---

4. THE PROPOSED APPROACH

In addition to separately encoding local features for every mention within a document as local models do, HEGEL constructs an informative heterogeneous graph for each document and applies a heterogeneous GNN, which encodes global coherence based on different information types. Finally, HEGEL combines local and global features to generate a final score for each mention-candidate pair.

Figure 2 [Figure 2: see original paper] provides an overview of HEGEL, which follows a four-stage processing pipeline: (a) encoding local features for each candidate independently, (b) informative graph construction for the document, (c) applying heterogeneous GNN on the graph, and (d) combining local and global features for scoring.

4.1 Encoding Local Features

Given a mention m_i in D and a candidate entity $e \in C_i$, HEGEL computes three types of local features to encode local mention-entity compatibility. These features consist of: (a) the Mention-Entity Prior $P(e|m)$, used in the candidate gen-

eration stage as described in Section 3.1; (b) the Context Similarity $\Psi_C(e, m_i)$, which utilizes an attention neural network to compute similarity between candidate e and local context words $w \in c_{m_i}$ surrounding m_i by selecting K most relevant words from c_{m_i}, eliminating noisy context words from computation:

$$\Psi_C(e, m_i) = \max_{w \in \text{topK}(c_{m_i})} \frac{v_e^T A v_w}{\|v_e\| \|v_w\|}$$

where v_e, v_w are entity embeddings and word embeddings trained in [**?**], and diagonal matrices A, B are both trainable; (c) the Type Similarity $\Psi_T(e, m_i)$, which estimates similarity between the types (PER, GPE, ORG, and UNK) of m_i and e by training a typing system proposed in [**?**]:

$$\Psi_T(e, m_i) = \frac{\text{Emb}_T(\text{Type}(m_i))^T \text{Emb}_T(\text{Type}(e))}{\|\text{Emb}_T(\text{Type}(m_i))\| \|\text{Emb}_T(\text{Type}(e))\|}$$

where $\text{Emb}_T(t)$ is a trainable type embedding for type t. As mentions and entities use the same embedding set, m_i and e with the same type will have higher Ψ_T than different types.

4.2 Informative Heterogeneous Graph Construction

For document D, HEGEL builds an informative heterogeneous graph G_D to collect different types of linking clues. As shown in Figure 2, $G_D = \langle V_D, E_D \rangle$ contains three node types: mention nodes V_{Ment}, entity nodes V_{Ent}, and keyword nodes V_{Word}. Therefore, the node set $V_D = V_{\text{Ment}} \cup V_{\text{Ent}} \cup V_{\text{Word}}$.

V_{Ment} naturally consists of all mentions m_i in D. V_{Ent} contains two parts of entities: the mention candidates $V_{\text{Ent},1} = \bigcup_i C_i$ where duplicates are removed, and the common neighbors in KB of at least two candidate entities in $V_{\text{Ent},1}$, formally $V_{\text{Ent},2} = \{v \in \text{KB} \mid \exists e_i, e_j \in V_{\text{Ent},1}, i \neq j, (e_i, v) \in \text{KB} \wedge (e_j, v) \in \text{KB}\}$. As reserving all neighbors in KB of $V_{\text{Ent},1}$ is computationally unacceptable, we eliminate nodes with only one neighbor in $V_{\text{Ent},1}$ from $V_{\text{Ent},2}$ because neighbors bridging two candidates are more informative for determining relations between candidates, as theoretically explained and experimentally proved in [**?**, **?**].

V_{Word} consists of keywords extracted from the Wikipedia page of each candidate in $V_{\text{Ent},1}$. We found that the first sentence on an entity's Wikipedia page usually contains more fine-grained type information, which is a very useful linking clue. Therefore, for $e \in V_{\text{Ent},1}$, we extract the first sentence s from its Wikipedia page, find the first link verb in s, and pick the continuous phrase immediately after the link verb, which contains only nouns, adjectives, and conjunctions. We regard words in the picked phrase, excluding stopwords, as keywords characterizing the fine-grained type of e and add them to V_{Word}.

After generating node set V_D, HEGEL creates heterogeneous edges between nodes of the same or different types following these rules: (a) edges between two

mention nodes $E_{MM} \subset V_{\text{Ment}} \times V_{\text{Ment}}$ are created between adjacent mentions (m_i, m_{i+1}) in D; (b) edges between two entity nodes $E_{EE} \subset V_{\text{Ent}} \times V_{\text{Ent}}$ are created when a relation exists between them in KB; (c) edges between two word nodes $E_{WW} \subset V_{\text{Word}} \times V_{\text{Word}}$ are created when the cosine similarity of two word embeddings exceeds a given threshold ϵ; (d) edges from entities to mentions $E_{EM} \subset V_{\text{Ent},1} \times V_{\text{Ment}}$ are consistent with the mention-candidate relation; (e) edges from words to entities $E_{WE} \subset V_{\text{Word}} \times V_{\text{Ent},1}$ are created when the word is one of the keywords for the entity. Note that (d) and (e) are uni-directional while (a)-(c) are bi-directional; the performance impact of constructing bi-directional edges for (d) and (e) will be discussed later. In short, the entire edge set can be represented as $E_D = E_{MM} \cup E_{EE} \cup E_{WW} \cup E_{EM} \cup E_{WE}$.

4.3 Heterogeneous Graph Neural Network

Given a constructed heterogeneous informative graph G_D, HEGEL applies a designed heterogeneous graph neural network (HGNN) to integrate different sources of manifold information and encourage interactions among them, generating information-augmented embeddings of V_{Ment} and $V_{\text{Ent},1}$ for later candidate scoring and ranking.

To avoid requiring expert knowledge and information loss from earlier metapath-based HGNN methods, we designed a novel metapath-free HGNN model. For heterogeneous graph G_D, we represent an edge $e \in E_D$ from node $i \in V_D$ to node $j \in V_D$ with edge type r as (i, j, r). Note that in our informative graph, the node types (t_i, t_j) can exclusively determine the edge type r, and therefore we denote (t_i, t_j) as r in the following explanation.

4.3.1 Node Embeddings For a mention node m_i, we use a text convolutional neural network (CNN) on the local context c_{m_i} to compute initial embeddings:

$$h^0_{m_i} = \text{CNN}(v_{w_1}, ..., v_{w_{\text{len}(c_{m_i})}})$$

where v_{w_i} are corresponding word embeddings of mention surface words and the mention' s context c_{m_i}, respectively, and $[\cdot;\cdot]$ is concatenation. For nodes in V_{Ent} and V_{Ment}, we naturally use entity/word embeddings R_e and R_w trained in [**?**] as initial embeddings h^0_e and $h^0_{m_i}$.

4.3.2 Inter-Node Propagation A node should receive different types of information from its heterogeneous neighborhood in different ways. Motivated by previous work on metapaths [**?**], HEGEL models different types of information propagation with multiple feature transformations on different adjacent relations. Considering edge type $r = (t_i, t_j)$, a node v_j with type t_j collects information from its neighborhood $N_{t_i}(v_j)$ with type t_i in the l-th layer by a Graph Convolutional Network (GCN):

$$h_{v_j,t_i}^{l+1} = \sigma\left(\sum_{v_i \in N_{t_i}(v_j)} \frac{1}{Z_{v_j,t_i}} W_{t_i,t_j}^{l} h_{v_i}^{l}\right)$$

where $h_{v_i}^{l} \in \mathbb{R}^{d_{l,t_i}}$ is v_i' s embedding before the l-th layer, $W_{t_i,t_j}^{l} \in \mathbb{R}^{d_{l+1,t_j} \times d_{l,t_i}}$ is a trainable matrix in the l-th layer, h_{v_j,t_i}^{l+1} is v_j' s new embedding related to t_i, and Z_{v_j,t_i} is the normalization factor. Note that for edge types (t_i, t_i) connecting nodes with the same type, self-loop connections are added to the edge set.

4.3.3 Intra-Node Aggregation To preserve information from different types of relationships with neighborhoods, for node v_j, HEGEL aggregates new embeddings to generate the input $h_{v_j}^{l+1}$ for the next layer:

$$h_{v_j}^{l+1} = \sigma\left(f_{\text{agg}}\left(\{h_{v_j,t_i}^{l+1} \mid t_i \in \{t \mid (t,t_j) \in R\}\}\right)\right)$$

where $f_{\text{agg}} : \mathbb{R}^{|\{t_i\}| \times d_{l+1,t_j}} \to \mathbb{R}^{d_{l+1,t_j}}$ is the aggregation function transforming $|\{t_i\}|$ input embeddings to an aggregated one, implemented as simple summation $f_{\text{agg}}(\{x_i\}) = \sum x_i$. σ is an activation function implemented as GELU$(\cdot)$ [**?**], and $h_{v_j}^{l+1}$ is the output embedding of the l-th layer containing all types of one-hop neighborhoods of v_j in the heterogeneous graph structure. As all neighborhood types can affect the current layer' s output and consequently information propagation in the next layer, we believe that by encouraging full interactions among different information types in this stage, the L layers of inter-node propagation and intra-node aggregation can encourage heterogeneous integrations and interactions among information types, represented by the final output.

4.3.4 Global Score Calculation After obtaining information-augmented embeddings $h_{m_i}^{L}$ for mention m_i and h_e^{L} for corresponding candidate e, HEGEL applies a bi-linear similarity calculation to represent global compatibility between the mention-candidate pair:

$$\Psi_{\text{Global}}(e, m_i) = (h_{m_i}^{L})^T M_{\text{Global}} h_e^{L}$$

where $M_{\text{Global}} \in \mathbb{R}^{d_{L,\text{Ment}} \times d_{L,\text{Ent}}}$ is a trainable diagonal matrix.

4.4 Feature Combining and Model Training

HEGEL combines local features and the global compatibility score to compute the linking score for each candidate e of mention m_i:

$$S(m_i, e) = f_{\text{comb}}([\Psi_{\text{Local}}(m_i, e); \Psi_{\text{Global}}(e, m_i)])$$

where f_{comb} is a two-layer fully connected neural network. The candidate e with the highest final linking score $S(m_i, e)$ in candidate list C_i is selected as the output linking result for m_i. HEGEL links m_i to NIL if and only if its candidate list $C_i = \emptyset$, meaning no corresponding entity exists for m_i in KB entity set E.

Following previous works, HEGEL attempts to make the ground truth entity e_i^* rank higher than other candidates, and therefore minimizes the following margin-based ranking loss:

$$\mathcal{L} = \sum_{m_i \in M} \sum_{e \in C_i \setminus \{e_i^*\}} \max(0, c - S(m_i, e_i^*) + S(m_i, e))$$

where $c > 0$ is the margin hyperparameter, and $[x]_+$ equals x when $x > 0$, or 0 otherwise.

5.1 Datasets

Following previous EL practice, we evaluated HEGEL on the benchmark dataset AIDA CoNLL-YAGO [**?**] for training, validation, and in-domain testing. To examine cross-domain generalization ability, we used five popular datasets for cross-domain testing: MSNBC [**?**], AQUAINT [**?**], ACE2004 [**?**], CWEB [**?**], and WIKIPEDIA [**?**]. Table 1 shows statistics and corresponding candidate generation recall for all datasets used in our experiments.

Table 1. The statistics of used datasets.
Dataset | #Mentions | #Docs | #Ments / #Docs | Recall (%)
AIDA-train (train) | | | |
AIDA-A (valid) | | | |
AIDA-B (test) | | | |
MSNBC | | | |
AQUAINT | | | |
ACE2004 | | | |
WIKIPEDIA | | | |

Note: Recall represents the ratio of ground truth entities appearing in generated candidate lists for corresponding mentions.

5.2 Model Variant

To examine our claim that the heterogeneous feature of GNN plays a crucial role in HEGEL, we implemented a semi-heterogeneous version called HEGEL-semi, which shares GCN parameters across different node types in every layer except the first, as input node embedding dimensions differ and cannot be processed in a non-heterogeneous way:

$$h_{v_j}^{l+1} = \sigma\left(\sum_{t_i \in \{t|(t,t_j)\in R\}} \sum_{v_i \in N_{t_i}(v_j)} \frac{1}{Z_{v_j,t_i}} W_{t_j}^{l} h_{v_i}^{l}\right)$$

As the $K-1$ parameter-sharing layers do not use different parameters to handle different node types, they do not benefit from heterogeneous graph structure. Therefore, HEGEL-semi' s performance should be lower than HEGEL according to our claim about heterogeneous method effects.

5.3 Experiment Settings

We used pre-trained Word2vec [**?**] word embeddings and entity embeddings released by [**?**], fixing embedding dimension $d_h = 300$. Hyperparameters were manually tuned based on validation performance on AIDA-A: CNN output dimension $d_{\text{cnn}} = 64$, all informative graph embedding dimensions $d_{l,t} = 32$ for $l = 1, \ldots, L$, number of HGNN layers $L = 2$, margin $c = 0.01$, $K = 40$, dropout rate 0.5, and E_{WW} threshold $\epsilon = 0.5$. To confine graph size within a computable range, documents with more than 80 mentions were split into several documents as evenly as possible.

We used Adam optimizer to train HEGEL with learning rate $a = 2 \times 10^{-4}$. The model was evaluated every 3 epochs, and training terminated when the highest validation performance was not exceeded for 10 evaluations. HEGEL-semi was implemented under the same settings as HEGEL due to achieving the best performance on AIDA-A.

5.4 Compared Baselines

To illustrate the effect of modeling interactions among different information types, we evaluated and compared HEGEL' s performance with nine existing methods on in-domain and cross-domain datasets:

- **AIDA** [**?**]: Builds a graph with coherent scores and similarities as weights, applying traditional statistical methods.
- **GLOW** [**?**]: Designs several statistical features for both local and global contexts using Wikipedia linking structure.
- **RI** [**?**]: Provides an Integer Linear Programming (ILP) formulation of Wikification incorporating entity-relation inference.
- **WNED** [**?**]: Builds disambiguation graphs and applies iterative random walks based on Information Theory.
- **Deep-ED** [**?**]: Leverages learned neural representations based on local context windows for joint document-level entity linking.
- **Ment-Norm** [**?**]: Treats and exploits relations between entities as latent variables based on Deep-ED.
- **GNED** [**?**]: Applies GCN and CRF on a homogeneous graph with extracted words and entities as nodes.

- **NCEL** [**?**]: Applies GCN on a bipartite graph to integrate both local contextual features and global information.
- **SGEL** [**?**]: Builds a graph for every mention sequentially, containing previously linked entities and candidates of unpredicated mentions.

Notably, GNED claims to first construct a heterogeneous entity-word graph to model global information, but their nodes are not truly heterogeneous as entity nodes share the same vector space with words. Additionally, they apply no heterogeneous architecture in their GNN, regarding all edges as the same type. Therefore, to our best knowledge, HEGEL is the first work to employ a heterogeneous GNN in EL tasks.

5.5 Main Results

We report performance of all compared baselines and our HEGEL in Table 2 . The top part shows non-GNN-based baselines; other baselines are GNN-based.

Table 2. Performance on in-domain (AIDA-B) and cross-domain datasets.

Models	In-domain	Cross-domain
	AIDA-B	MSNBC
Prior p(e	m)	
Deep-ED		
Ment-Norm		
HEGEL - w/o VWord		
- w/o VEnt,2		
HEGEL-semi		
Local		
HEGEL		

Note: We show in-KB accuracy (%) for in-domain datasets and micro-F1 score (%) for cross-domain datasets. For HEGEL we show std. deviation obtained over 3 runs.

The in-domain test dataset AIDA-B, which shares similar data distribution with training dataset AIDA-train and validation dataset AIDA-A, is the most important benchmark. By modeling latent relations between mentions and injecting entity coherence—which can be regarded as simple interaction between two information types—Ment-Norm outperforms all baselines on AIDA-B, showing that interactions of heterogeneous information benefit global coherence capture. We observed that HEGEL, which integrates manifold linking knowledge in a more interactive and effective way for capturing global coherence, significantly outperforms Ment-Norm, demonstrating that HEGEL can encourage richer interactions among different information types and greatly improve performance.

No model consistently achieves the best F1-score across all five cross-domain datasets. HEGEL outperforms the other two GNN-based methods, NCEL and

SGEL, on MSNBC and ACE2004, showing that HEGEL handles cross-domain linking cases better than them to some extent.

HEGEL performs extremely well on in-domain cases by making full use of different linking clues for better capturing global coherence, but shows no advantage on cross-domain datasets. We found that ground truth entities in cross-domain test sets are less popular, where linking clues are sparse. To improve generalization on such tough cases, the effective approach appears to be introducing large-scale corpus for training to "see" linking clues of cross-domain entities at training time. We will attempt to introduce large-scale pre-trained language models like BERT to improve HEGEL' s generalization ability in future work.

As HEGEL-semi is also implemented with $L = 2$, it contains one heterogeneous layer and one parameter-sharing layer. Results in Table 2 confirm that although HEGEL-semi outperforms the local model, its lack of heterogeneous information propagation in the second layer leads to obvious performance drops compared to HEGEL, demonstrating that heterogeneous GNN is important for HEGEL' s strong performance.

Comparing GNED with our simpler, effective keyword extraction method from the first Wikipedia sentence, GNED searches the entire Wikipedia KB to find hyperlinks to corresponding entities and extracts contexts during preprocessing, requiring iteration through all $|E|$ entities and becoming very time-consuming. Even with less keyword evidence, our strategy still outperforms GNED on in-domain datasets with lower time overhead. GNED accesses more additional linking clues and achieves better performance on cross-domain datasets; we suppose richer information could also improve HEGEL' s generalization ability and further boost cross-domain performance.

5.6 Ablation Study

As shown in the bottom part of Table 2, HEGEL boosts local model performance with an average improvement of 1.77%, demonstrating HEGEL' s ability to greatly enhance local models.

To further examine our heterogeneous model' s effect, we removed keyword nodes V_{Word} and neighbor nodes $V_{\text{Ent},2}$ from V_D, respectively, along with related edges from E_D. This caused significant performance drops (0.89% and 0.61% on average, respectively) across datasets, especially in-domain AIDA-B (1.71% and 1.43%). Results demonstrate the effectiveness of introducing keyword (fine-grained type) information and neighborhood information of candidate entities and modeling interactions among them, which helps accurately capture topical characteristics of candidates.

5.7 Analysis

5.7.1 The Impact of Edge Directions

As mentioned in Section 4.2, HEGEL only keeps one direction for E_{EM} and E_{WE}. We hypothesize that adding edges from V_{Ment} to $V_{\text{Ent},1}$ and from $V_{\text{Ent},1}$ to V_{Word} would lead to over-smoothing, as candidates to be disambiguated relate to the same mention and possibly the same keywords, where they might entangle and make disambiguation harder. As expected, results in Table 3 prove that keeping these edges uni-directional alleviates over-smoothing and enhances performance.

Table 3. Experiment results on changing edge directionality.

Models	AIDA-B	Cross-domain avg.
HEGEL		
+ VEnt → VWord		
+ VMent → VEnt		
+Both		

5.7.2 The Impact of the Number of GNN Layers

Despite GNN' s powerful ability to process graph-structured data, most are shallow, meaning they have few propagation layers. As shown in [**?**], stacking many layers with non-linear functions degrades GNN-based model performance due to over-smoothing. Therefore, we examined HEGEL' s performance with different numbers of layers. Results in Figure 3 [Figure 3: see original paper] agree with previous GNN works: HEGEL with $K = 2$ layers achieves the best performance in EL tasks. Too many layers cause over-smoothing, and a 1-layer model is insufficient to propagate heterogeneous information required for aggregation and interaction on the graph.

To alleviate over-smoothing when training deeper GNNs, residual connection [**?**] is used between GNN hidden layers as a variant to facilitate information retention through deeper models [**?**]. Residual connection enables HEGEL to carry over heterogeneous information from previous layer input embeddings by modifying Equation (14):

$$h_{v_j}^{l+1} = \sigma\left(f_{\text{agg}}(\{h_{v_j,t_i}^{l+1}\}) + h_{v_j}^{l}\right)$$

However, as shown in Figure 3, applying residual connection to the model with $K = 2$ causes drops in both in-domain and cross-domain performance. Though residual connection boosts in-domain performance for $K \geq 3$, results remain incomparable to the best $K = 2$ performance. We hypothesize this relates to varying information handling methods across HGNN layers, as heterogeneous structures in different propagation steps are too different to be correctly handled by the same network layer.

5.7.3 Error Analysis

We randomly sampled and analyzed 100 mentions from all mentions incorrectly linked by HEGEL from in-domain dataset AIDA-B and the most difficult cross-domain dataset CWEB. As shown in Table 4 , four major error types include: (1) **Topic Errors**, occurring when HEGEL links candidates of different (usually unrelated) topics to gold entities, representing the main challenge for current global methods; (2) **Similar Entity Errors**, where predicted and gold entities have too similar semantics for disambiguation by local and global information, potentially solvable by introducing more information; (3) **Related Entity Errors**, occurring when predicted entities are semantically closely related to gold ones, such as a city and stadium located within it or a hypernym of the gold entity; (4) **Dataset Annotation Errors**, where gold entities in datasets are wrong and differ from predictions, occurring only in CWEB.

Table 4. Major error types and examples.

Error types	Examples
Topic ErrorsAIDA-B: 24%CWEB: 44%	···To win [Timbo]' s trust, Chris chained himself up in the elephant' s enclosure ··· HEGEL→Timbo (a town in Guinea)Gold→Timbaland (an American Musician)
Similar Entity ErrorsAIDA-B: 29%CWEB: 24%	···In a gloomy Geneva conference centre built before the dawn of the [Internet], groups of staid officials made a ··· HEGEL→Internet (the worldwide computer network)Gold→World Wide Web (the global system of pages via URL)
Related Entity ErrorsAIDA-B: 47%CWEB: 29%	···a small rightwing [Christian] civil war militia, Saqr, whose trial was concurrent ··· HEGEL→ChristianGold→Catholicism (the largest Christian church)
Dataset Annotation ErrorsAIDA-B: 0%CWEB: 3%	···Brooks Cole Herring. [B.], 2001, Ethical guidelines in the treatment of compulsive ··· HEGEL→B.W. Aston (a Texas historian and professor)Gold→B (the second letter ?)

Note: Square brackets denote target mentions. Italicized and underlined entities

are HEGEL predictions and dataset gold entities, respectively.

5.7.4 Case Study

As shown in Figure 2, HEGEL must map mentions "Scotland," "Murrayfield," "Cuttitta," and "England" in the same document to corresponding entities. "Murrayfield" and "Cuttitta" are unambiguous with only one candidate each. However, "Scotland" and "England" are linked to wrong candidates by the local model, while HEGEL outputs correct answers by properly modeling interactions among heterogeneous information types, especially from the neighborhood around "Marcello Cuttitta" (a former rugby union player) and "Rugby Union," and from respective keywords related to "rugby."

Ablation score calculations in Table 5 demonstrate that information from keyword nodes V_{Word} and neighbor nodes $V_{\text{Ent},2}$ and correct information handling are both important for HEGEL to properly capture topical coherence and model heterogeneous interactions.

Table 5. Scores in case study.

Models	Scot.→country	Scot.→team	Eng.→football	Eng.→rugby
HEGEL				
- VWord				
- VEnt,2				

6. CONCLUSION AND FUTURE WORK

We presented HEGEL, a novel graph-based global entity linking method designed to model and utilize interactions among heterogeneous information types from different sources. We achieved this by constructing a document-level informative heterogeneous graph and applying a heterogeneous GNN to propagate and aggregate information on the graph, which is difficult for previous homogeneous architectures. Extensive experiments on standard benchmarks show HEGEL achieves state-of-the-art performance in EL tasks.

ACKNOWLEDGEMENTS
This work is supported in part by the National Key R&D Program of China (No. 2020AAA0106600) and the Key Laboratory of Science, Technology and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology).

AUTHOR CONTRIBUTIONS
Z.B. Chen (czb-peking@pku.edu.cn) performed the research, designed the

methodology, conducted experiments and analysis, and wrote the manuscript. Y.T. Wu (wyting@pku.edu.cn) proposed research problems, designed experiments and analysis, and wrote the manuscript. Y.S. Feng (fengyansong@pku.edu.cn) and D.Y. Zhao (zhaodongyan@pku.edu.cn) proposed research problems, supervised the research, and provided insightful manuscript revisions. All authors made valuable contributions.

REFERENCES

[1] Yih, S. W.t., et al.: Semantic parsing via staged query graph generation: Question answering with knowledge base. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 1321–1331 (2015)

[2] Wang, J., et al.: Combining knowledge with deep convolutional neural networks for short text classification. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI), pp. 2915–2921 (2017)

[3] Hoffmann, R., et al.: Knowledge-based weak supervision for information extraction of overlapping relations. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 541–550 (2011)

[4] Luan, Y., et al.: Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. arXiv preprint arXiv:1808.09602 (2018)

[5] Cao, Y., et al.: Neural collective entity linking. arXiv preprint arXiv:1811.08603 (2018)

[6] Fang, Z., et al.: Joint entity linking with deep reinforcement learning. In: The World Wide Web Conference, pp. 438–447 (2019)

[7] Fang, Z., et al.: High quality candidate generation and sequential graph attention network for entity linking. In: Proceedings of the Web Conference 2020, pp. 640–650 (2020)

[8] Ganea, O.E., Hofmann, T.: Deep joint entity disambiguation with local neural attention. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 2619–2629 (2017)

[9] Guo, Z., Barbosa, D.: Robust named entity disambiguation with random walks. Semantic Web 9, 1–21 (2017)

[10] Hong, H., et al.: An attention-based graph neural network for heterogeneous structural learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 4132–4139 (2020)

[11] Le, P., Titov, I.: Improving entity linking by modeling latent relations between mentions. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1595–1604 (2018)

[12] Liu, M., et al.: A multi-view-based collective entity linking method. ACM Transactions on Information Systems 37(2), 1–29 (2019)

[13] Ratinov, L., et al.: Local and global algorithms for disambiguation to Wikipedia. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 1375–1384 (2011)

[14] Wu, J., et al.: Dynamic graph convolutional networks for entity linking. In: Proceedings of the Web Conference 2020, pp. 1149–1159 (2020)

[15] Xu, P., Barbosa, D.: Neural fine-grained entity type classification with hierarchy-aware loss. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 16–25 (2018)

[16] Hu, L., et al.: Graph neural entity disambiguation. Knowledge-Based Systems 195, 105620 (2020)

[17] Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)

[18] Cheng, X., Roth, D.: Relational inference for Wikification. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013), pp. 1787–1796 (2013)

[19] Hoffart, J., et al.: Robust disambiguation of named entities in text. In: Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, pp. 782–792 (2011)

[20] Mulang, I.O., et al.: Evaluating the impact of knowledge graph context on entity disambiguation models. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 2157–2160 (2020)

[21] Raiman, J., Raiman, O.: Deeptype: Multilingual entity linking by neural type system evolution. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 5406-5413 (2018)

[22] Wu, L., et al.: Zero-shot entity linking with dense entity retrieval. arXiv preprint arxiv:1911.03814 (2019)

[23] Cao, N.D., et al.: Multilingual autoregressive entity linking. arXiv preprint arxiv:2103.12528 (2021)

[24] Hamilton, W.L., Ying, R., Leskovec, J.: Inductive representation learning on large graphs. In: Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS' 17), pp. 1025–1035 (2017)

[25] Veličković, P., et al.: Graph attention networks. In: International Conference on Learning Representations (ICLR), pp. 1-12 (2018)

[26] Dong, Y., Chawla, N.V., Swami, A.: metapath2vec: Scalable representation learning for heterogeneous networks. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, pp. 135–144 (2017)

[27] Moreau, E., Yvon, F., Cappé, O.: Robust similarity measures for named entities matching. In: Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pp. 593–600 (2008)

[28] Hendrycks, D., Gimpel, K.: Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415 (2016)

[29] Cucerzan, S.: Large-scale named entity disambiguation based on Wikipedia data. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pp. 708–716 (2007)

[30] Milne, D., Witten, I.H.: Learning to link with Wikipedia. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM ' 08), pp. 509–518 (2008)

[31] Le, Q., Mikolov, T.: Distributed representations of sentences and documents. In: International Conference on Machine Learning (PMLR), pp. 1188-1196 (2014)

[32] Chen, M., et al.: Simple and deep graph convolutional networks. In: International Conference on Machine Learning (PMLR), pp. 1725–1735 (2020)

[33] He, K., et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770-778 (2016)

AUTHOR BIOGRAPHY

Zhibin Chen is a graduate student at the Academy for Advanced Interdisciplinary Studies (AAIS), Peking University, China. He works in the Web Information Processing (WIP) group of Wangxuan Institute of Computer Technology (WICT), Peking University. His research interests focus on natural language processing, information extraction, and graph learning. ORCID: 0000-0003-1140-9808

Yuting Wu is a Ph.D. student at Peking University. She works with Prof. Dongyan Zhao and Prof. Yansong Feng in the Web Information Processing (WIP) group of Wangxuan Institute of Computer Technology (WICT), Peking University. Her research interests include knowledge fusion/linking

(knowledge graph embedding, entity alignment, entity linking, etc.) and graph neural networks. ORCID: 0000-0002-7550-3804

Yansong Feng is an Associate Professor in the Wangxuan Institute of Computer Technology (WICT) at Peking University, affiliated with the Web Information Processing group. Previously, he worked with Prof. Mirella Lapata and obtained his Ph.D. from the School of Informatics at the University of Edinburgh. Prior to Edinburgh, he worked with Prof. Jufu Feng on pattern recognition at Peking University. His current research focuses on distilling knowledge from large volumes of Web text resources.

Dongyan Zhao is a Professor in Wangxuan Institute of Computer Technology (WICT), Peking University, China. He received B.S., M.S., and Ph.D. degrees in Computer Science from Peking University' s Department of Computer Science and Technology. As a distinguished member of China Computer Federation (CCF), he served as secretary-general of CCF TCCI (Technical Committee on Chinese Information Technology, renamed Technical Committee on Natural Language Processing in 2020) from 2010-2019, and is a member of CCF Task Force on Big Data and CCF Network and Data Communications, as well as a senior member of CIPS Social Media Processing Committee.

Note: Figure translations are in progress. See original paper for figures.