HEER: Heterogeneous Graph Embedding for Emerging Relation Detection from News

Jingyuan Zhang*, Chun-Ta Lu*, Mianwei Zhou†, Sihong Xie‡, Yi Chang† and Philip S. Yu*§
*Department of Computer Science, University of Illinois at Chicago, IL, USA
†Yahoo! Research, Sunnyvale, CA, USA
‡Computer Science and Engineering Department, Lehigh University, PA, USA
§Institute for Data Science, Tsinghua University, Beijing, China
{jzhan8, clu29, psyu}@uic.edu, sxie@cse.lehigh.edu, {mianwei, yichang}@yahoo-inc.com

***Abstract*—Real-world knowledge is growing rapidly nowadays. New entities arise with time, resulting in large volumes of relations that do not exist in current knowledge graphs (KGs). These relations containing at least one new entity are called *emerging relations*. They often appear in news, and hence the latest information about new entities and relations can be learned from news timely. In this paper, we focus on the problem of discovering emerging relations from news. However, there are several challenges for this task: (1) at the beginning, there is little information for emerging relations, causing problems for traditional sentence-based models; (2) no negative relations exist in KGs, creating difficulties in utilizing only positive cases for emerging relation detection from news; and (3) new relations emerge rapidly, making it necessary to keep KGs up to date with the latest emerging relations. In order to address these issues, we start from a global graph perspective and propose a novel Heterogeneous graph Embedding framework for Emerging Relation detection (HEER) that learns a classifier from positive and unlabeled instances by utilizing information from both news and KGs. Furthermore, we implement HEER in an incremental manner to timely update KGs with the latest detected emerging relations. Extensive experiments on real-world news datasets demonstrate the effectiveness of the proposed HEER model.**

***Keywords*-Heterogeneous Networks; Emerging Relations; Embedding;**

I. Introduction

A *relation* is about the connection between two entities. Entities can be persons, organizations, locations, etc., and examples of relations can be person-affiliation and organization-location. Recognizing relations between entities is required in a lot of real-world applications in information extraction, natural language understanding and information retrieval. Hence, extracting relations from unstructured texts, such as newswire, blogs, and so on have received considerable attention in the last few years [1–4].

Conventional approaches of relation extraction from texts focus on a local view, where each sentence mentioning two entities is considered for feature learning. For example, the works in [1–3] extracted a lot of sentence-level features including lexical part-of-speech tags of words, syntactical dependency tree paths, etc.. In order to achieve good performance, such local view based methods require large amounts of sentences to extract useful sentence-level features. For those relations with few sentences, these methods would be problematic.

Some other works attempt to leverage knowledge graphs (KGs), such as Freebase[1] and DBpedia[2], to provide useful supervisions for extracting relations from texts. For instance, [2] uses relation instances in Freebase instead of annotated texts as their source of supervision. However, these methods are handicapped due to the limited coverage of existing KGs [5]. As shown in [6], 71% of the roughly 3 million people in Freebase have no known place of birth, 94% have no known parents, and 99% have no known ethnicity. Therefore, a lot of research work tries to fill in the missing relations to mitigate the problem of knowledge sparsity [7–13]. For example, Path Ranking Algorithm (PRA) [14, 15] performs link prediction in KGs via a random walk inference technique; embedded representations of entities and relations in KGs are learned to infer missing relations in [16–18].

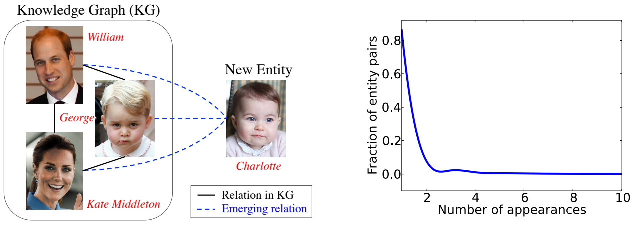
Nowadays, real-world knowledge is growing rapidly. New entities arise with time [19], resulting in large volumes of relations that do not exist in current KGs. We call such relations *emerging relations*[3]. Emerging relations often appear in news, and hence the latest information about new entities and relations can be learned from news timely. For example, when a new baby (e.g., Charlotte in Figure 1(a)) is born in the Royal Family, no information about this baby exists in KGs. However, there are lots of news talking about this new baby and her family. Therefore, the relation between the new baby and her parent is an emerging relation and it can be detected from news.

In this paper, we study the problem of discovering emerging relations from news. Detecting such relations has many benefits to real-world applications. Emerging relations can help expand current KGs and keep them up to date. In addition, emerging relations can also help news related tasks,

[1] https://www.freebase.com/

[2] http://wiki.dbpedia.org/

[3] The relations with both entities in KGs are out of scope of this paper since they can be inferred via the existing KG completion methods.



(a) Real-world examples (b) Appearance distributions

Figure 1. Examples of emerging relations and appearance distributions of emerging relations.

such as news retrieval and ranking, event detection, etc.. However, detecting emerging relations is a challenging task due to the following reasons:

- Sentence-level features for emerging relations are usually rare. A data analysis conducted on 6 million online news headlines from Yahoo!, as shown in Figure 1(b), reveals that 86% of emerging entity pairs appear in only one sentence. Simply relying on sentence-level features extracted from few sentences could lead to sub-optimal results for emerging relation detection.
- Due to the lack of negative relations in KGs, previous methods [2, 10, 14] often apply different strategies to extract negative relations. However, the negative relations could be false negative [2] in reality, which may introduce noise and cause degraded performance for emerging relation detection.
- With massive amounts of news arriving every second, new relations emerge rapidly. It is necessary to keep KGs up to date with the latest emerging relations.

In order to address these issues, we start from a global graph perspective instead of the traditional local sentence perspective and propose a novel **H**eterogeneous graph **E**mbedding framework for **E**merging **R**elation detection (HEER). Figure 2 shows the simplified procedure of HEER with an example. To capture the global corpus information in news, HEER constructs a heterogeneous textual graph from news. Two kinds of nodes – entities and contextual words – are involved in the graph and the link between two nodes represents their co-occurrence statistics from the whole news corpus. By jointly learning from the heterogeneous textual graph and the knowledge graph, HEER can embed words and entities into a low dimensional space. These graph-based embeddings not only preserve the semantic relatedness of entities and contextual words, but also have a strong predictive power for the detection of emerging relations. To deal with the lack of negative relations in reality, HEER further predicts the emerging relations via a positive and unlabeled learning (PU) classifier [20] on the embeddings of entities.

In summary, our contributions are as follows:

- We define *emerging relations* and propose a HEER framework to detect emerging relations by utilizing information from both news and KGs.

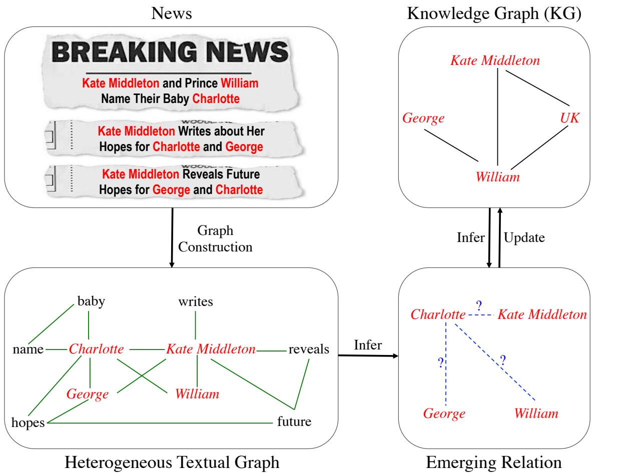


Figure 2. Detecting emerging relations by inferring from the heterogeneous textual graph and the KG. The entities are in red. The co-occurrence links in the heterogeneous textual graph are in green and the relations in KG are in black.

- We learn a classifier based on positive and unlabeled instances in the proposed HEER method by taking advantage of existing relations in KGs.
- We further implement HEER in an incremental manner to timely update KGs with the latest detected emerging relations.
- We conduct extensive empirical studies on real-world news data to demonstrate the effectiveness of the proposed HEER method.

The rest of the paper is organized as follows. Section II formulates the problem; Section III introduces the details of the proposed HEER framework; Section IV presents the experimental setup and the results; Section V briefly reviews related work; and Section VI concludes this study.

II. Preliminary

In this paper, we study the problem of discovering emerging relations from news. Before proceeding, we introduce the related concepts and notations.

Definition 1: **Entity and Relation:** An entity e_i can represent a person, an organization, or a location, etc.. We use $y \in \{0, 1\}$ to denote the binary relation label for an entity pair (e_i, e_j). If two entities e_i and e_j have a relation in reality, such as a person-affiliation and an organization-location relation, $y = 1$. Otherwise, $y = 0$.

Definition 2: **Knowledge Graph (KG):** A *knowledge graph* is denoted as an undirected graph $\mathcal{G}_{kg} = (E_{kg}, \mathcal{E}_{kg})$, which keeps the known relations between entity pairs. For an entity pair (e_i, e_j), we use a KG label z to show whether the entity pair can match with a relation in the given KG or not, i.e., $z = 1$ if $(e_i, e_j) \in \mathcal{E}_{kg}$. Otherwise, $z = 0$.

Because of the limited coverage of the existing KG [6], an entity pair with a KG label of $z = 0$ does not mean there is no relation for this pair (i.e., $y = 0$). Since the exact relation labels for entity pairs without KG labels are unknown to us, we call them *unlabeled relations*.

Nowadays, due to the rapid growth of real-world knowledge, large volumes of *emerging relations* are arising with time. An emerging relation is defined as follows:

Definition 3: **Emerging Relation:** An emerging relation between an entity pair (e_i, e_j) exists, if its relation label $y = 1$ and it contains at least one entity that is not included in the given KG (i.e., $e_i \notin E_{kg}$ or $e_j \notin E_{kg}$).

For example, in Figure 1(a), (*Charlotte, Kate Middleton*) is an emerging relation since *Charlotte* is a new entity and she is a child of *Kate Middleton*. Similarly, (*Charlotte, William*) and (*Charlotte, George*) are also examples of emerging relations. For an emerging relation, its KG label z always equals to 0 because at least one entity is not included in the KG.

Our goal is to learn the relation labels for those emerging entity pairs when the news just start talking about them. With rare sentences about emerging entity pairs, the traditional sentence-based methods could lead to sub-optimal results for emerging relation detection. Compared to traditional local sentence based relation detection, we construct a heterogeneous textual graph from news to capture the global corpus information in news.

Definition 4: **Heterogeneous Textual Graph:** A heterogeneous textual graph is represented as an undirected graph $\mathcal{G}_{news} = (\mathcal{V}_{news}, \mathcal{E}_{news})$. $\mathcal{V}_{news}$ is the set of nodes (objects), including two types of objects, i.e., entities $E_{news} = \{e_1, ..., e_M\}$ and contextual words $C_{news} = \{c_1, ..., c_N\}$. $\mathcal{E}_{news} \subseteq \mathcal{V}_{news} \times \mathcal{V}_{news}$ is the set of links (edges) between the nodes in $\mathcal{V}_{news}$, which involves the links of entity-entity, entity-word, and word-word co-occurrences.

An example of the heterogeneous textual graph is shown in Figure 2. Each link in the graph represents the co-occurrence of two nodes in news sentences and its weight equals to the frequencies of co-occurrences of these two nodes. For instance, the link between *Charlotte* and *baby* shows that these two nodes co-appear in some news sentence and the weight of this link is 1 since these two nodes co-appear in the first news only.

Such a heterogeneous textual graph helps encode the global corpus information in news. Besides, the existing KG provides helpful guidance for learning relations between entity pairs. We can utilize the heterogeneous textual graph and the current KG together for detecting emerging relations. However, it is challenging since entities associated with emerging relations are missing in current KGs. In addition, no negative relations exist in KGs, creating difficulties in utilizing only positive and unlabeled instances.

In order to address these challenges, we propose a novel **H**eterogeneous graph **E**mbedding framework for **E**merging **R**elation detection (HEER).

III. Proposed Method

In this section, we first introduce how HEER constructs a heterogeneous textual graph from news. Then we describe how HEER can jointly learn from the heterogeneous textual graph and the existing KG to embed every entity and contextual word into a low dimensional space. After that, we present the learning classifier with only positive and unlabeled relations. Furthermore, we discuss how to implement HEER in an incremental manner to timely update the KG with the latest emerging relations.

A. Constructing a Heterogeneous Textual Graph from News

Given a large collection of news $\mathcal{D}$, the proposed HEER method first extracts entities and contextual words to build the heterogeneous textual graph. In this paper, entities in news are annotated with the Stanford Named Entity Recognizer (NER) tool[4]. We mainly focus on 3 types of entities, namely person, location and organization, and consider the public available DBpedia dataset as the given knowledge graph. The entities that cannot be exactly matched to DBpedia are viewed as new entities. Excluding entities, the remaining uni-gram words in news are considered as contextual words and we remove stop words beforehand. Each entity and each uni-gram contextual word are nodes in the constructed heterogeneous textual graph. In order to extract the co-occurrence links in the graph, nodes within every 5-word sliding window in a news sentence are considered to be co-occurring with each other as in [21]. We use the frequencies of nodes co-appearing in news sentences as weights of corresponding links.

B. Joint Embedding of the News and the KG

Given the constructed heterogeneous textual graph and the KG, we aim to learn a low dimensional space for every entity and contextual word. The learned embeddings should not only fit the relation information in the KG, but also reflect the text descriptions of emerging relations in news. In order to achieve this, we should jointly embed the heterogeneous textual graph and the KG. In the following, we first explain how to learn graph embeddings from a single graph, and then present how to jointly embed multiple graphs.

According to the types of links, the heterogeneous textual graph $\mathcal{G}_{news}$ can be split into three sub-graphs: the homogeneous entity-entity sub-graph $\mathcal{G}_{ee} = (E_{news}, \mathcal{E}_{ee})$, the bipartite entity-word sub-graph $\mathcal{G}_{ec} = (\mathcal{V}_{news}, \mathcal{E}_{ec})$ and the homogeneous word-word sub-graph $\mathcal{G}_{cc} = (C_{news}, \mathcal{E}_{cc})$. These three sub-graphs together capture the global corpus information in news. Given the bipartite entity-word sub-graph $\mathcal{G}_{ec} = (\mathcal{V}_{news}, \mathcal{E}_{ec})$, for instance, we aim to embed each entity $e_i \in E_{news}$ and each word $c_j \in C_{news}$ into low-dimensional vectors $\mathbf{s}_i \in \mathbb{R}^d$ and $\mathbf{t}_j \in \mathbb{R}^d$. Here d is the dimension of embedding vectors and $d \ll |\mathcal{V}_{news}|$.

[4] http://nlp.stanford.edu/software/CRF-NER.shtml

In order to learn the embeddings, for each co-occurrence link $(e_i, c_j) \in \mathcal{E}_{ec}$, we first define the conditional probability of e_i given c_j as

$$P(e_i|c_j) = \frac{e^{\mathbf{s}_i^\top \mathbf{t}_j}}{\sum_{k=1}^{|E_{news}|} e^{\mathbf{s}_k^\top \mathbf{t}_j}}. \tag{1}$$

For each word c_j, this probability actually calculates a conditional distribution $P(\cdot|c_j)$ over all the entities in E_{news}. In the low-dimensional space, we intend to preserve the *second-order proximity*, which means two nodes are similar to each other if they have similar neighbors [22], by making $P(\cdot|c_j)$ be close to its empirical distribution $\hat{P}(\cdot|c_j)$. Here we define $\hat{P}(\cdot|c_j) = \frac{w_{j\cdot}}{o_j}$, where $w_{j\cdot}$ is the weight of the edge $(e_\cdot, c_j)$ and o_j is the sum of weights for edges connected to c_j, i.e., $o_j = \sum_{e_k \in N(c_j)} w_{jk}$, where $N(c_j)$ is the set of entity neighbors of c_j.

By minimizing the Kullback-Leibler (KL) divergence between two distributions $\hat{P}(\cdot|c_j)$ and $P(\cdot|c_j)$, we obtain the objective function for embedding the bipartite entity-word sub-graph $\mathcal{G}_{ec}$ as follows:

$$J_{ec} = -\sum\nolimits_{(e_i,c_j)\in\mathcal{E}_{ec}} w_{ij} \log P(e_i|c_j). \tag{2}$$

However, it is time-consuming to directly optimize Equation (2) since it requires to sum over the entire set of links when calculating the conditional probability $P(\cdot|c_j)$. In order to address this issue, we adopt the techniques of negative sampling [21], where for each edge selected with a probability proportional to its weight, multiple negative links (edges) are sampled from some noisy distribution. For the detailed optimization process, readers can refer to [21].

Since a homogeneous graph can be easily converted to a bipartite graph, we can derive similar objective functions for embedding the entity-entity sub-graph $\mathcal{G}_{ee}$ and the word-word sub-graph $\mathcal{G}_{cc}$ as follows:

$$J_{ee} = -\sum\nolimits_{(e_i,e_j)\in\mathcal{E}_{ee}} w_{ij} \log P(e_i|e_j), \tag{3}$$

$$J_{cc} = -\sum\nolimits_{(c_i,c_j)\in\mathcal{E}_{cc}} w_{ij} \log P(c_i|c_j). \tag{4}$$

With the objectives (2), (3) and (4), we can learn vector representations of the heterogeneous textual graph:

$$J_{news} = J_{ec} + J_{ee} + J_{cc}. \tag{5}$$

Besides the heterogeneous textual graph, the current KG contains a large amount of positive relations between entities, providing helpful guidance for learning relations between entities. Since the KG is a homogeneous entity-entity graph $\mathcal{G}_{kg}$ about real-world relations, we can learn vector representations of the KG in a similar way:

$$J_{kg} = -\sum\nolimits_{(e_i,e_j)\in\mathcal{E}_{kg}} w_{ij} \log P(e_i|e_j), \tag{6}$$

where $\mathcal{E}_{kg}$ is the set of positive relations in the KG and we set the weight w_{ij} for each relation as 1. In order to learn from both the news and the KG, we combine them together as the final objective of the proposed HEER model:

$$J = \theta J_{kg} + (1-\theta) J_{news}. \tag{7}$$

Here $\theta \in [0, 1]$ is a guiding parameter that trades off between news and the KG. Specifically, $\theta = 0$ (or 1) indicates that only the news (or the KG) is utilized in learning embeddings. In addition, a higher θ indicates that the KG plays a more important role in the process of embedding.

Algorithm 1 Joint embedding of the news and the KG

```
Input: The heterogeneous textual graph $\mathcal{G}_{news} = \mathcal{G}_{ec} \cup \mathcal{G}_{ee} \cup \mathcal{G}_{cc}$,
    the KG $\mathcal{G}_{kg}$, the guiding parameter $\theta$, the number of negative
    samples $k$, and the number of embedding iterations $T$.
Output: Entity embeddings $\mathcal{S}$ and word embeddings $\mathcal{T}$.
Initialize entity embeddings $\mathcal{S}$ randomly
Initialize contextual word embeddings $\mathcal{T}$ randomly
while iter $\leq T$ do
    Generate a random number $\gamma \in [0, 1]$
    if $\gamma \leq \theta$ then
        EMBEDDING UPDATE($\mathcal{S}$, $\mathcal{S}$, $\mathcal{G}_{kg}$, $k$)
    else
        EMBEDDING UPDATE($\mathcal{S}$, $\mathcal{T}$, $\mathcal{G}_{ec}$, $k$)
        EMBEDDING UPDATE($\mathcal{S}$, $\mathcal{S}$, $\mathcal{G}_{ee}$, $k$)
        EMBEDDING UPDATE($\mathcal{T}$, $\mathcal{T}$, $\mathcal{G}_{cc}$, $k$)
    end if
end while

function EMBEDDING UPDATE($\mathcal{S}$, $\mathcal{T}$, $\mathcal{G}$, $k$)
    Sample an edge from $\mathcal{G}$ and draw $k$ negative edges
    Update node embeddings $\mathcal{S}$ and $\mathcal{T}$
end function
```

Since the links in different sub-graphs have different meanings in reality, we sample links from each sub-graph independently to optimize Equation (7). The detailed process of the graph embedding is summarized in Algorithm 1.

C. Detecting Emerging Relations with Positive Cases Only

After the graph embedding procedure, each entity and word can be represented with a d-dimensional vector. We use such global graph-based embeddings of entities as features for emerging relation detection. Due to the lack of negative relations in the KG, it is challenging to detect emerging relations with positive instances only. In the following, we present a positive and unlabeled (PU) learning classifier to address this issue.

Given an instance of an entity pair (e_i, e_j), we can represent the feature of the instance as $\mathbf{x} = h(\mathbf{s}_i, \mathbf{s}_j)$, where h is a function of the entity embedding vectors $\mathbf{s}_i$ and $\mathbf{s}_j$. Different formulations of h can be derived to represent the pair features, such as the concatenation, the average, etc.. In this paper, we simply take $h(\mathbf{s}_i, \mathbf{s}_j) = \frac{1}{2}(\mathbf{s}_i + \mathbf{s}_j)$. Let $\mathbf{X} = \{\mathbf{x} : (e_i, e_j) \in \mathcal{E}_{ee}\}$ denote the feature representations of all the entity pairs co-occurring in the news. We consider $\mathbf{X}$ as the input feature matrix of the PU classifier for emerging relation detection.

As mentioned in Section II, each entity pair has a KG label z showing whether an entity pair can match with a relation in KG (i.e., $z = 1$) or not (i.e., $z = 0$). For the positive entity pairs $\mathcal{P}$ with KG labels of $z = 1$, we denote their feature matrix as $\mathbf{X}(\mathcal{P})$. The feature matrix of the remaining unlabeled entity pairs $\mathcal{U}$ with KG labels of $z = 0$ are denoted as $\mathbf{X}(\mathcal{U})$. We further denote the emerging entity pairs as $\mathcal{L}$ and their feature matrix as $\mathbf{X}(\mathcal{L})$. Here $\mathcal{L}$ is a subset of $\mathcal{U}$. We use $\mathbf{y}$ and $\mathbf{z}$ to denote the relation labels and KG labels for all the entity pairs in $\mathcal{P} \cup \mathcal{U}$. Our ultimate goal is to predict the relation labels $\mathbf{y}$ for emerging entity pairs $\mathcal{L}$ by learning from $\mathcal{P}$ and $\mathcal{U}$.

For entity pairs in $\mathcal{U}$, the relation labels $\mathbf{y}$ are unknown but the KG labels $\mathbf{z}$ are known. Hence, we propose to train a PU classifier f on $\mathcal{P} \cup \mathcal{U}$ to learn the relation labels $\mathbf{y}$ by inferring from the KG labels $\mathbf{z}$. We adopt the idea from [20] to adjust a classifier g on KG labels $\mathbf{z}$ to a classifier f on relation labels $\mathbf{y}$ with a constant factor.

We first train a standard classifier g on KG labels $\mathbf{z}$ that yields a function $g(\mathbf{x}) = P(z = 1|\mathbf{x})$ for each instance of an entity pair in $\mathcal{P} \cup \mathcal{U}$. Here $P(z = 1|\mathbf{x})$ is the probability that an instance with feature $\mathbf{x}$ has a positive KG label, i.e., $z = 1$. By assuming that entity pairs in $\mathcal{P}$ are chosen completely randomly from all real relations in $\mathcal{P} \cup \mathcal{U}$, we can show that $P(z = 1|\mathbf{x}, y = 1) = P(z = 1|y = 1)$. This is the classic "selected at random" assumption in [20] and it is proved that

$$P(y = 1|\mathbf{x}) = \frac{P(z = 1|\mathbf{x})}{P(z = 1|y = 1)}. \tag{8}$$

Equation (8) shows that we can predict the probability $P(y = 1|\mathbf{x})$ of the relation label by estimating $P(z = 1|y = 1)$, which is the probability that an entity pair with relation label $y = 1$ exists in the KG, i.e., $z = 1$. Here $P(z = 1|y = 1)$ can be effectively estimated by using the classifier g and a set of entity pairs S randomly sampled from $\mathcal{P} \cup \mathcal{U}$. Let S_p be the subset of entity pairs in S with positive KG labels (i.e., $z = 1$) and $\mathbf{X}(S_p)$ is the corresponding feature set. We can obtain the following formula:

$$P(z = 1|y = 1) \sim \varepsilon = \frac{1}{|S_p|} \sum\nolimits_{\mathbf{x} \in \mathbf{X}(S_p)} g(\mathbf{x}), \tag{9}$$

where the estimator ε is the average value of $g(\mathbf{x})$ for $\mathbf{x}$ in $\mathbf{X}(S_p)$. Since ε is based on a certain number of data instances, it has a low variance and is preferable in practice [20]. With ε and the classifier g on KG labels $\mathbf{z}$, we can adjust to a classifier f on relation labels $\mathbf{y}$ as follows:

$$f(\mathbf{x}) = P(y = 1|\mathbf{x}) = \frac{g(\mathbf{x})}{\varepsilon}. \tag{10}$$

Figure 3 shows the learning process of the PU classifier on positive relations $\mathcal{P}$ and unlabeled relations $\mathcal{U}$. Since the classifier g considers all unlabeled relations as negative, an entity pair with a relation label $y = 1$ may be wrongly predicted as negative if it has a KG label $z = 0$. By adjusting g to f with an estimator ε, we can learn the correct relation label for an entity pair with a KG label of $z = 0$. After training, we can predict the relation labels for emerging entity pairs $\mathcal{L}$. By integrating the graph construction, the graph embedding, and the positive-unlabeled learning together, the proposed HEER framework can detect emerging relations from news effectively. We summarize the whole process of HEER in Algorithm 2.

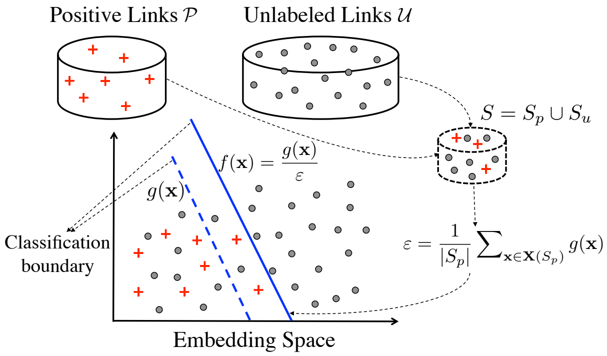

Positive Links $\mathcal{P}$
Unlabeled Links $\mathcal{U}$
$S = S_p \cup S_u$
$f(\mathbf{x}) = \frac{g(\mathbf{x})}{\varepsilon}$
$g(\mathbf{x})$
Classification boundary
$\varepsilon = \frac{1}{|S_p|} \sum_{\mathbf{x} \in \mathbf{X}(S_p)} g(\mathbf{x})$
Embedding Space

Figure 3. The PU learning classifier in HEER. g is a classifier on KG labels $\mathbf{z}$. f is the PU classifier on relation labels $\mathbf{y}$. By estimating a constant factor ε from some sampled entity pairs S, we can adjust g to f on relation labels $\mathbf{y}$.

Algorithm 2 The HEER algorithm

```
Input: A set of news texts $\mathcal{D}$ and the KG $\mathcal{G}_{kg}$.
Output: A PU learning classifier $f$ for emerging entity pairs $\mathcal{L}$.
    //constructing graphs
    Extract entities and contextual words from $\mathcal{D}$
    Construct the heterogeneous textual graph $\mathcal{G}_{news}$
    //joint embedding
    Learn embeddings according to Algorithm 1
    //detecting emerging relations
    Learn the feature set $\mathbf{X}$ after embedding
    Get positive and unlabeled relations $\mathcal{P}$ and $\mathcal{U}$
    Train a classifier $g$ on KG labels $\mathbf{z}$
    Estimate $\varepsilon$ from $g$
    Learn $f$ using $g$ and $\varepsilon$ on relation labels $\mathbf{y}$
    Predict relations labels for emerging entity pairs $\mathcal{L}$
```

D. Incremental Update of the KG

As news arrives rapidly with huge amounts of emerging relations, it is essential to timely update the KG with the latest emerging relations. In this section, we show how to implement HEER in an incremental manner for KG updates.

Assume we collect news at regular time intervals, e.g., one day, one week or one month. All the news texts at current time t are $\mathcal{D}_t$ and the current KG is $\mathcal{G}_{kg}^t$. The proposed HEER can learn entity embeddings $\mathcal{S}_t$ and contextual word embeddings $\mathcal{T}_t$ from both $\mathcal{D}_t$ and $\mathcal{G}_{kg}^t$. We denote the embedding results as $\{\mathcal{S}_t, \mathcal{T}_t\}$. At time $t+1$, the news will be updated as $\mathcal{D}_{t+1}$, including $\mathcal{D}_t$ and the newly arrived news between time t and $t+1$. Note that there is no need to retrain the embeddings from scratch. Instead, we can

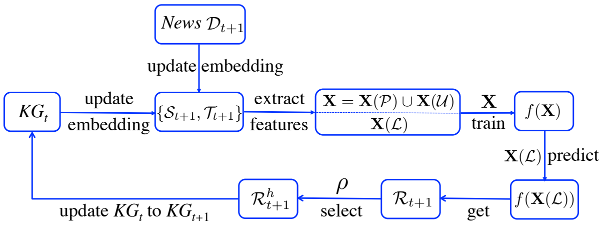

News $\mathcal{D}_{t+1}$
update embedding
KG_t
update
embedding
$\{\mathcal{S}_{t+1}, \mathcal{T}_{t+1}\}$
extract
features
$\mathbf{X} = \mathbf{X}(\mathcal{P}) \cup \mathbf{X}(\mathcal{U})$
$\mathbf{X}(\mathcal{L})$
$\mathbf{X}$
train
$f(\mathbf{X})$
$\mathbf{X}(\mathcal{L})$ predict
$f(\mathbf{X}(\mathcal{L}))$
get
$\mathcal{R}_{t+1}$
ρ
select
$\mathcal{R}^h_{t+1}$
update KG_t to KG_{t+1}

Figure 4. The update procedure of the incremental HEER. At time $t+1$, the embeddings $\mathcal{S}$ and $\mathcal{T}$ are first updated by learning from the newly arrived news and the KG. Then the features $\mathcal{X}$ are extracted based on the updated embeddings. After the training of the PU learning classifier f on $\mathbf{X}$, the latest emerging relations $\mathcal{R}_{t+1}$ will be detected from emerging entity pairs $\mathcal{L}$. At last, those highly-reliable relation $\mathcal{R}^h_{t+1}$ will be added into the KG.

reuse the previous trained embeddings $\{\mathcal{S}_t, \mathcal{T}_t\}$ at time t as initialization in Algorithm 1 to learn $\{\mathcal{S}_{t+1}, \mathcal{T}_{t+1}\}$ at time $t+1$.

With the updated embeddings, HEER can detect a set of emerging relations $\mathcal{R}_{t+1}$ from $\mathcal{D}_{t+1}$. In order to keep the KG up to date, we should add $\mathcal{R}_{t+1}$ into $\mathcal{G}^t_{kg}$. However, there are some false positive relations in $\mathcal{R}_{t+1}$. If we add all the detected relations into the KG, it will increase the noisy information and reduce the quality of the KG. Therefore, we select those highly-reliable emerging relations from $\mathcal{R}_{t+1}$ with a threshold ρ. Specifically, given the feature set $\mathbf{x}_{t+1}$ for a detected relation in $\mathcal{R}_{t+1}$, if the PU learning probability $f(\mathbf{x}_{t+1}) \geq \rho$, we will add the relation into $\mathcal{G}^t_{kg}$. Otherwise, we will discard it. We denote these highly-reliable emerging relations as $\mathcal{R}^h_{t+1}$. They can be further considered as positive instances during the PU learning process. With these highly-reliable emerging relations, we can get a new KG $\mathcal{G}^{t+1}_{kg}$. The update procedure of the incremental HEER is illustrated in Figure 4.

IV. Experiments

In this section, we conduct extensive experiments to evaluate the proposed HEER framework. After introducing the datasets and the experimental settings, we compare different baseline methods.

A. Data Processing

Two real-world news datasets are used in the experiment.

• **Yahoo! News**[5]: We collect a large set of online English news from Yahoo! News in October 2015. Only the headline information is considered for Yahoo! News dataset.

• **BBC News**[6] [23]: Documents in five topical areas are collected from the BBC news website from 2004 to 2005. We consider each sentence in the document as a piece of news.

We annotate entities and contextual words according to the method in Section III-A. In order to find new entities and emerging relations, we map the entities to the knowledge graph DBpedia with exact match. The entities that cannot be matched are considered as new entities. To avoid the high-cost of human labeling, we focus on the existing relations in DBpedia to evaluate the effectiveness of the proposed HEER framework.

For all the entities E_{news} in the existing relations, we randomly select half of them as new entities. We denote these new entities as E_n and the remaining half as E_o. The entity pairs with KG labels of $z = 1$ and with both entities in E_o are regarded as positive instances, denoted as $\mathcal{P}$. The entity pairs with KG labels of $z = 1$ and with at least one entity in E_n are regarded as emerging relations. Since the emerging relations are unlabeled positive instances, we denote them as $\mathcal{U}_p$. The entity pairs with KG labels of $z = 0$ and with both entities in E_{news} are regarded as unlabeled negative instances, denoted as $\mathcal{U}_n$. Thus, the unlabeled instances, denoted as $\mathcal{U}$, are the union of positive and negative unlabeled instances, i.e. $\mathcal{U} = \mathcal{U}_p \cup \mathcal{U}_n$. The statistics of the two datasets are summarized in Table I.

B. Compared Methods

In order to show that the HEER model can effectively detect emerging relations, we compare the following methods.

- BOW: It is based on the classical "bag-of-words" representation. Each entity pair is represented with a $|C_{news}|$-dimensional vector, in which the weight of each dimension is calculated by the number of times the word and the entity pair co-occur in news.
- LS: It is the standard **L**ocal **S**entence based classifier (LS) [2] using a variety of sentence-level features, including lexical part-of-speech tags of words and syntactical dependency tree paths.
- SG: It is based on the state-of-the-art word embedding model, **S**kip-**G**ram (SG) [21]. It learns embedding vectors for each word in news, where each entity is consider as a word in this paper.
- DW: It is based on the **D**eep**W**alk model (DW) [24]. DW is only applicable for homogeneous graphs with binary edges. It learns embeddings of nodes by applying truncated random walks on the graph. By viewing entities and contextual words as one type of node, we can build a homogeneous graph from news and apply DW on this graph.
- LINE: It is based on the **L**arge-scale **I**nformation **N**etwork **E**mbedding method (LINE) [22]. Similar to DW, LINE treats entities and contextual words as one type of nodes but considers the weights of the edges when learning the embeddings.
- PTE: It is based on the **P**redictive **T**ext **E**mbedding method (PTE) [25]. It learns embeddings of nodes from the heterogeneous textual graph and the KG. The tradeoff between news and the KG is not considered in PTE.

[5] https://www.dropbox.com/s/yad2tfaj9ve3vuf/yahoo_news_titles.tar.gz?dl=0

[6] http://mlg.ucd.ie/files/datasets/bbc-fulltext.zip

Table I
STATISTICS OF THE DATASETS.

	News	Heterogeneous textual graph					Knowledge Graph		Classification instances		
Dataset	$\|\mathcal{D}\|$	$\|E_{news}\|$	$\|C_{news}\|$	$\|\mathcal{E}_{ee}\|$	$\|\mathcal{E}_{ec}\|$	$\|\mathcal{E}_{cc}\|$	$\|E_{kg}\|$	$\|\mathcal{E}_{kg}\|$	$\|\mathcal{P}\|$	$\|\mathcal{U}_p\|$	$\|\mathcal{U}_n\|$
Yahoo!	6,209,256	13,801	61,705	20,136	398,466	697,804	22,157	710,994	3,297	9,246	12,543
BBC	44,088	2,556	7,273	873	19,206	57,373	2,030	43,689	167	575	742

Table II
THE CLASSIFICATION PERFORMANCE "AVERAGE SCORE±STD (RANK)" ON EMERGING RELATION DETECTION TASK. "↑" INDICATES THE LARGER THE VALUE THE BETTER THE PERFORMANCE.

(a) Results on the Yahoo! News dataset.

	Criteria		
Methods	AUC ↑	Accuracy ↑	F1 ↑
BOW	0.562±0.006 (10)	0.510±0.001 (11)	0.118±0.002 (10)
LS	0.653±0.005 (4)	0.506±0.002 (12)	0.026±0.005 (12)
SG	0.547±0.006 (12)	0.543±0.004 (7)	0.188±0.014 (8)
DW	0.587±0.010 (8)	0.516±0.001 (10)	0.064±0.004 (11)
LINE	0.600±0.012 (6)	0.533±0.003 (9)	0.129±0.011 (9)
PTE	0.732±0.008 (3)	0.502±0.000 (13)	0.010±0.002 (13)
BOW-PU	0.559±0.006 (11)	0.541±0.007 (8)	0.567±0.011 (3)
LS-PU	0.647±0.002 (5)	0.610±0.002 (4)	0.481±0.011 (7)
SG-PU	0.545±0.007 (13)	0.610±0.007 (4)	0.517±0.008 (6)
DW-PU	0.587±0.009 (8)	0.577±0.006 (6)	0.545±0.008 (4)
LINE-PU	0.598±0.011 (7)	0.617±0.008 (3)	0.538±0.008 (5)
PTE-PU	0.734±0.007 (2)	0.675±0.006 (2)	0.671±0.006 (2)
HEER	0.786±0.007 (1)	0.717±0.008 (1)	0.716±0.005 (1)

(b) Results on the BBC News dataset.

	Criteria		
Methods	AUC ↑	Accuracy ↑	F1 ↑
BOW	0.552±0.028 (9)	0.496±0.013 (13)	0.053±0.034 (9)
LS	0.632±0.028 (4)	0.501±0.003 (11)	0.005±0.011 (12)
SG	0.571±0.009 (6)	0.520±0.005 (7)	0.127±0.021 (8)
DW	0.516±0.022 (13)	0.506±0.004 (9)	0.034±0.017 (10)
LINE	0.538±0.035 (10)	0.505±0.005 (10)	0.029±0.017 (11)
PTE	0.664±0.029 (2)	0.500±0.001 (12)	0.001±0.001 (13)
BOW-PU	0.560±0.023 (8)	0.544±0.019 (5)	0.531±0.045 (5)
LS-PU	0.624±0.027 (5)	0.606±0.022 (2)	0.468±0.051 (7)
SG-PU	0.571±0.019 (6)	0.598±0.017 (4)	0.553±0.026 (3)
DW-PU	0.520±0.026 (12)	0.517±0.027 (8)	0.520±0.023 (6)
LINE-PU	0.536±0.039 (11)	0.537±0.037 (6)	0.541±0.039 (4)
PTE-PU	0.656±0.017 (3)	0.601±0.014 (3)	0.639±0.016 (2)
HEER	0.712±0.033 (1)	0.644±0.035 (1)	0.649±0.024 (1)

Table III
EXAMPLES OF EMERGING RELATIONS DETECTED FROM YAHOO! NEWS. THE TAG "(IN)" AND "(OUT)" NEXT TO AN ENTITY INDICATES WHETHER THIS ENTITY IS IN THE KG OR NOT.

Relation		News text
Elizabeth McGovern (out)	Downton Abbey (out)	1. Elizabeth McGovern: Broadway trip by Downton Abbey cast was like being a Beatle.
Alcatel-Lucent (out)	Michel Combes (in)	1. Alcatel-Lucent slashes payout to former chief Michel Combes.
Saudi Arabia (out)	Shaybah (out)	1. Saudi Arabia to invest US$45 billion in Shaybah oil field expansion.
Bernie Sanders (in)	University of Chicago (out)	1. Bernie Sanders speaks at the University of Chicago. 2. Bernie Sanders to speak at University of Chicago Monday. 3. Bernie Sanders calls on students to join his fight during University of Chicago stop.
Ghana (out)	Kwame Nkrumah (out)	1. Ghana celebrates Dr. Kwame Nkrumah today.
David Helfgott (in)	Melbourne (out)	1. David Helfgott, Australia's most well-known classical pianist, is coming to Melbourne.

All the above baseline methods train a standard classifier by treating all the unlabeled instances as negative ones. These baselines can be used to show the effectiveness of the positive-unlabeled (PU) learning for the detection of emerging relations from news. In addition, the PU learning on different kinds of feature sets are compared:

- BOW-PU: It is the **BOW** based **PU** classifier (BOW-PU). We apply the PU learning on the BOW features.
- LS-PU: It is the **L**ocal **S**entence based **PU** classifier (LS-PU). We train a PU classifier on the local sentence features.
- SG-PU: It is the **S**kip-**G**ram based **PU** classifier (SG-PU). After SG gets the embeddings, the PU learning algorithm is applied.
- DW-PU: It is the **D**eep**W**alk based **PU** classifier (DW-PU). After DW obtains the embeddings, the PU learning algorithm is applied.
- LINE-PU: It is the **LINE** based **PU** classifier (LINE-PU). After LINE learns the embeddings, the PU learning algorithm is applied.
- PTE-PU: It is the **PTE** based **PU** classifier (PTE-PU). After PTE learns the embeddings, the PU learning algorithm is applied.
- HEER: It is the **H**eterogeneous graph **E**mbedding for **E**merging **R**elation detection (HEER) proposed in this paper. We apply the PU learning after embedding the heterogeneous textual graph and the KG.

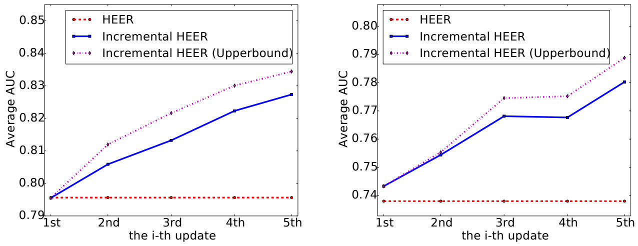
For fair comparisons, random forest classifier is used for all the above approaches and the number of trees in the forest is set as 100. The number of BOW features is 61,705 and 7,273 for Yahoo! News and BBC News, respectively. The number of sentence-level features is 286,461 and 21,481 for Yahoo! News and BBC News, respectively. For the embedding methods, the dimensionality of the embeddings is set to 50 and the average embedding of each entity pair is the input feature of the classifier. The guiding parameter θ of HEER is set to 0.2 for both datasets. Other settings of the graph embedding are the same as in [22, 25]. When estimating ε in the PU learning, as in [20], 10% of instances are used for the estimation.

To evaluate the performance of the compared approaches, we randomly sample 80% of instances in $\mathcal{U}$ and keep all the instances in $\mathcal{P}$ as the training set, and use the remaining 20% of instances in $\mathcal{U}$ as the testing set. This random sampling experiment is repeated 5 times. We consider the AUC, accuracy and F1 score as the evaluation metrics.

The average performance with the rank of each method is reported in Table II. It can be observed that PU learning methods perform much better than the standard positive-negative learning methods on the F1 metric. In addition, the proposed HEER consistently outperforms other PU baseline methods on both datasets.

We can find that the baselines without PU learning have similar accuracies but SG performs best on both datasets. It indicates the embedding methods can perform better than the traditional BOW and sentence-level features. However, since these baselines simply treat the unlabeled instances as negative ones, they perform worse than the PU learning methods for the task of emerging relation detection.

Among the baselines using the PU learning technique, we can observe that LS-PU performs worst on F1 although the feature size of LS-PU is the largest. The reason is that LS-PU cannot capture enough information for the classification task due to the sparsity of the sentence-level features. The recall of LS-PU is very low. However, the proposed HEER method captures global corpus information in news by building a heterogeneous textual graph. Therefore, with only 50 embedding features, HEER can have a much higher recall than LS-PU, thereby achieving a higher F1 score. Another discovery is that the homogeneous graph embedding models DW-PU and LINE-PU do not perform well on both datasets because they do not take the heterogeneity of news into account. In addition, the news and the KG are two heterogeneous data sources. If we simply combine them together as a homogeneous graph, DW-PU and LINE-PU will perform much worse though more information in the KG is considered. Hence we only show the better performing version of DW-PU and LINE-PU without using the KG in Table II. Furthermore, PTE-PU performs well on both datasets because it considers the heterogeneity in news and the heterogeneity between news and the KG. However, HEER outperforms PTE-PU because it considers the trade-offs between news and the KG.



(a) Yahoo! News (b) BBC News

Figure 5. The performance of the incremental HEER.

In summary, with the heterogeneous textual graph modeling, embedding and PU learning, the proposed HEER outperforms the baseline methods for both datasets.

C. Case Study

In this subsection, we present a case study to further show the effectiveness of the proposed HEER framework. We focus on the Yahoo! News dataset and show several examples of the discovered emerging relations in Table III. These relations are detected only by HEER and all the other baselines fail to discover these relations. It can be observed that most of these relations appear only once in Yahoo! News. For example, there is only one piece of news talking about the entity pair (*Elizabeth McGovern, Downton Abbey*). When there are rare appearances of the emerging relations, the word-level based and the sentence-level based models cannot extract sufficient features. Therefore, they cannot detect these emerging relations effectively. Since the proposed HEER framework builds a heterogeneous textual graph from news, it can capture the global corpus information for emerging relations with few sentences.

D. Incremental HEER

When we incrementally update the KG with the detected emerging relations, HEER can make use of an up-to-date KG during the embedding process to discover the latest emerging relations more accurately. However, the incremental HEER method may add some false positive relations into the KG and reduce the quality of the KG . With more incremental updates, the noisy information may be accumulated and the error rate of HEER may increase. In this subsection, we analyze how HEER performs on the two datasets when we incrementally update the KG with some false positive emerging relations.

We take the KG and 50% of news to pre-train the embeddings and split the remaining news randomly into five equal chunks for testing the incremental HEER. During each incremental update procedure, we detect the emerging relations on one chunk and update the KG with those highly-reliable emerging relations. In the incremental HEER, we would select highly-reliable emerging relations to update the

KG. In the experiment, we set the threshold $\rho = 0.95$ to select those highly-reliable emerging relations.

In order to evaluate the effectiveness of the proposed incremental HEER, two baselines are compared. One baseline is the HEER method without updating the KG. We just test the performance of HEER on each chunk using the model trained on the original 50% of news. The other one is the incremental HEER method that updates the KG with all the true positive emerging relations in the given chunk. Actually this is the optimal method to update the KG. It can show how the false positive emerging relations will affect the results. Since the order of chunks may influence the updating performance, in the experiment, we repeat 5 times with different orders of chunks. We only report the average AUC scores in Figure 5 because we have similar observations in other metrics.

It can be observed that the incremental HEER always outperforms HEER on both datasets no matter how many updates we do. With more updates, the performance is improving because more information is utilized to detect emerging relations. We can also observe that the difference between the incremental HEER and the optimal method is quite small, indicating that the added false positive emerging relations have little influence on the performance during each incremental update.

E. Parameter Analysis

In the proposed HEER framework, θ controls the relative importance of the KG guidance in detecting emerging relations. Here we assess the benefit of HEER with different values of the guiding parameter θ. Figure 6 represents the classification AUC scores and accuracies on both datasets. We observe that HEER can discover emerging relations more effectively with the help of the KG. Particularly, when there is no guidance of the KG, i.e., $\theta = 0$, HEER can perform well but not the best. When we increase θ, the performance becomes best for Yahoo! News when θ is around 0.2. For BBC News, $\theta \in [0.1, 0.4]$ yields similar good results. Figure 6 also shows that it is much easier to detect emerging relations from news than from the KG, because the performance drops when we increase θ from 0.5 to 1.

We also demonstrate the effect of the dimensionality of embeddings by fixing the other parameters. As shown in Figure 7, the best accuracy is achieved when the embedding dimension is 50 for both datasets. We can observe that the performance of HEER fluctuates a lot on BBC News. The reason may be that Yahoo! News is a larger dataset and HEER can be more stable on a larger dataset. We further analyze the performance of HEER with different numbers of iteration T. Figure 8 reports the average AUC and accuracy results. It can be seen that HEER converges after around 200 iterations on both datasets.

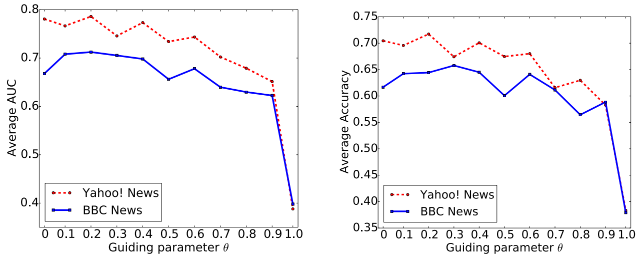


Figure 6. The performance with different guiding parameters.

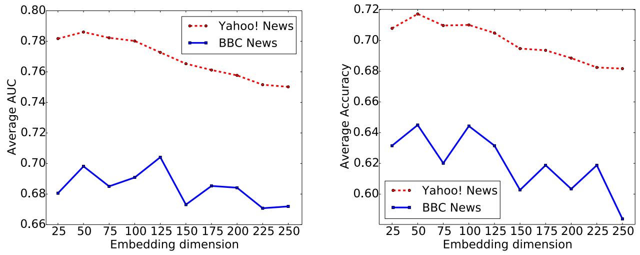


Figure 7. The performance with different embedding dimensions.

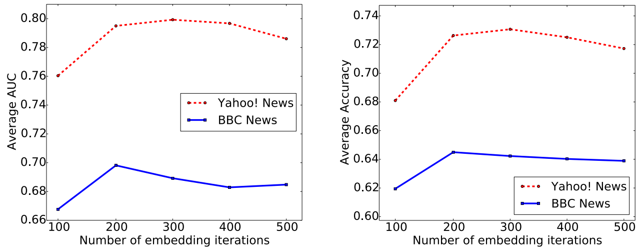


Figure 8. The performance with different embedding iterations.

V. Related Work

Relation extraction from texts has been well studied in recent years [1–3, 26, 27]. For example, a distantly-supervised learning is performed in [2] by using relation instances in the knowledge graph of Freebase instead of annotated texts as their source of supervision. It extracts a large amount of sentence-level features including lexical part-of-speech tags of words, syntactical dependency tree paths, etc.. However, there are few sentences about the emerging relations at the beginning. Simply relying on sentence-level features could lead to sub-optimal results for emerging relation detection.

Due to the limited coverage of the existing knowledge graph (KG) [5], the task of KG completion has received a lot of attention [8, 10, 11, 13, 28]. There are two branches for this task. One is to learn embedding representations of entities and relations in the KG and use these embeddings to infer missing relations [7, 12, 16–18]. The other branch is to predict missing relations from a graph view [9, 14, 15, 29]. For instance, the Path Ranking Algorithm (PRA) [14, 15] performs link prediction in the KG via a random walk inference technique. In addition, the research work [29] combines methods of the above two branches by using a recursive neural network to create embedded representations of PRA-style paths. In our work, the emerging relations have new entities that are not included in the KG. Hence, it is impossible to apply these techniques directly. Furthermore, some work tries to embed the knowledge graph and the texts jointly [30]. Our work is different since we focus on the emerging relations from news and model the news into a heterogeneous textual graph. We further update the KG incrementally with the detected emerging relations.

Our work is also related to the problem of information network modeling and mining [31, 32]. Recently, there are some work attempting to embed very large real-world networks [22, 24, 25]. DeepWalk and LINE are proposed in [24] and [22], respectively. These two models can only handle homogeneous networks. PTE is proposed in [25] to deal with heterogeneous networks.

The positive-unlabeled (PU) learning techniques have been proposed for many years [20, 33, 34]. We apply the PU learning technique proposed in [20] for the detection of emerging relations since it can easily adjust a standard positive-negative classifier to a positive-unlabeled classifier with a constant factor.

VI. Conclusion

In this paper, we define a new concept of "emerging relations" from news and focus on the problem of discovering such relations. We propose a novel **H**eterogeneous graph **E**mbedding framework for **E**merging **R**elation detection (HEER) that learns a classifier from positive and unlabeled instances by utilizing information from both news and the knowledge graph (KG). We show that by modeling news into a heterogeneous textual graph, the proposed method HEER can detect emerging relations effectively. We further implement HEER in an incremental manner to timely update the KG with the latest detected emerging relations.

Acknowledgement

This work is supported in part by NSF through grants IIS-1526499, and CNS-1626432. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X GPU used for this research.

References

[1] R. Hoffmann, C. Zhang, X. Ling, L. Zettlemoyer, and D. Weld, "Knowledge-based weak supervision for information extraction of overlapping relations," in *ACL-HLT*, 2011.

[2] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, "Distant supervision for relation extraction without labeled data," in *ACL-IJCNLP*, 2009.

[3] M. Surdeanu, J. Tibshirani, R. Nallapati, and C. Manning, "Multi-instance multi-label learning for relation extraction," in *EMNLP*, 2012.

[4] D. Zeng, K. Liu, S. Lai, G. Zhou, and J. Zhao, "Relation classification via convolutional deep neural network," in *COLING*, 2014.

[5] X. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun, and W. Zhang, "Knowledge vault: A web-scale approach to probabilistic knowledge fusion," in *KDD*, 2014.

[6] R. West, E. Gabrilovich, K. Murphy, S. Sun, R. Gupta, and D. Lin, "Knowledge base completion via search-based question answering," in *WWW*, 2014.

[7] K. Chang, W. Yih, B. Yang, and C. Meek, "Typed tensor decomposition of knowledge bases for relation extraction," in *EMNLP*, 2014.

[8] A. García-Durán, A. Bordes, and N. Usunier, "Effective blending of two and three-way interactions for modeling multi-relational data," in *ECML/PKDD*, 2014.

[9] M. Gardner, P. Talukdar, B. Kisiel, and T. Mitchell, "Improving learning and inference in a large knowledge-base using latent syntactic cues," in *EMNLP*, 2013.

[10] M. Gardner, P. Talukdar, J. Krishnamurthy, and T. Mitchell, "Incorporating vector space similarity in random walk inference over knowledge bases," in *EMNLP*, 2014.

[11] M. Nickel, X. Jiang, and V. Tresp, "Reducing the rank in relational factorization models by including observable patterns," in *NIPS*, 2014.

[12] R. Socher, D. Chen, C. Manning, and A. Ng, "Reasoning with neural tensor networks for knowledge base completion," in *NIPS*, 2013.

[13] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes," in *AAAI*, 2014.

[14] M. Gardner and T. Mitchell, "Efficient and expressive knowledge base completion using subgraph feature extraction," in *EMNLP*, 2015.

[15] N. Lao, T. Mitchell, and W. Cohen, "Random walk inference and learning in a large scale knowledge base," in *EMNLP*, 2011.

[16] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *NIPS*, 2013.

[17] A. Bordes, J. Weston, R. Collobert, and Y. Bengio, "Learning structured embeddings of knowledge bases," in *AAAI*, 2011.

[18] G. Ji, S. He, L. Xu, K. Liu, and J. Zhao, "Knowledge graph embedding via dynamic mapping matrix," in *ACL*, 2015.

[19] G. Weikum, S. Bedathur, and R. Schenkel, "Temporal knowledge for timely intelligence," in *Enabling Real-Time Business Intelligence*. Springer, 2011.

[20] C. Elkan and K. Noto, "Learning classifiers from only positive and unlabeled data," in *KDD*, 2008.

[21] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *NIPS*, 2013.

[22] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *WWW*, 2015.

[23] D. Greene and P. Cunningham, "Practical solutions to the problem of diagonal dominance in kernel document clustering," in *ICML*, 2006.

[24] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *KDD*, 2014.

[25] J. Tang, M. Qu, and Q. Mei, "Pte: Predictive text embedding through large-scale heterogeneous text networks," in *KDD*, 2015.

[26] G. Angeli, J. Tibshirani, J. Wu, and C. Manning, "Combining distant and partial supervision for relation extraction," in *EMNLP*, 2014.

[27] D. Zeng, K. Liu, Y. Chen, and J. Zhao, "Distant supervision for relation extraction via piecewise convolutional neural networks," in *EMNLP*, 2015.

[28] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion." in *AAAI*, 2015.

[29] A. Neelakantan, B. Roth, and A. McCallum, "Compositional vector space models for knowledge base completion," in *ACL*, 2015.

[30] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph and text jointly embedding," in *EMNLP*, 2014.

[31] J. Zhang, X. Kong, L. Jie, Y. Chang, and P. Yu, "Ncr: A scalable network-based approach to co-ranking in question-and-answer sites," in *CIKM*, 2014.

[32] J. Zhang, L. Jie, A. Rahman, S. Xie, Y. Chang, and P. Yu, "Learning entity types from query logs via graph-based modeling," in *CIKM*, 2015.

[33] B. Liu, Y. Dai, X. Li, W. Lee, and P. Yu, "Building text classifiers using positive and unlabeled examples," in *ICDM*, 2003.

[34] J. Zhang, P. Yu, and Z. Zhou, "Meta-path based multi-network collective link prediction," in *KDD*, 2014.