

## Research Article

# Link Trustworthiness Evaluation over Multiple Heterogeneous Information Networks

Meng Wang <sup>1,2</sup>, Xu Qin <sup>1,3</sup>, Wei Jiang <sup>4,5</sup>, Chunshu Li <sup>1</sup> and Guilin Qi <sup>1,2</sup>

<sup>1</sup>School of Computer Science and Engineering, Southeast University, Nanjing 210096, China

<sup>2</sup>Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, Nanjing 211189, China

<sup>3</sup>Southeast University-Monash University Joint Research Institute, Suzhou 215123, China

<sup>4</sup>Laboratory for Complex Systems Simulation, Institute of Systems Engineering, AMS, Beijing 100083, China

<sup>5</sup>North China Institute of Computing Technology, Beijing 100083, China

Correspondence should be addressed to Meng Wang; [meng.wang@seu.edu.cn](mailto:meng.wang@seu.edu.cn)

Received 5 December 2020; Revised 16 February 2021; Accepted 25 February 2021; Published 8 March 2021

Academic Editor: Jianxin Li

Copyright © 2021 Meng Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Link trustworthiness evaluation is a crucial task for information networks to evaluate the probability of a link being true in a heterogeneous information network (HIN). This task can significantly influence the effectiveness of downstream analysis. However, the performance of existing evaluation methods is limited, as they can only utilize incomplete or one-sided information from a single HIN. To address this problem, we propose a novel multi-HIN link trustworthiness evaluation model that leverages information across multiple related HINs to accomplish link trustworthiness evaluation tasks inherently and efficiently. We present an effective method to evaluate and select informative pairs across HINs and an integrated training procedure to balance inner-HIN and inter-HIN trustworthiness. Experiments on a real-world dataset demonstrate that our proposed model outperforms baseline methods and achieves the best accuracy and  $F1$ -score in downstream tasks of HINs.

## 1. Introduction

Heterogeneous information network (HIN) is an efficient technique to model complicated real-world information through a network structure, in which network nodes indicate multi-typed real-world nodes and edges indicate relationships between nodes. For example, MovieLens [1] and IMDB are typical examples of HIN in movie domains modeling relationships among movies, actors, directors, etc. Containing richer structural information, HINs bring about plenty of opportunities for data mining in complicated applications. However, erroneous information commonly exists in HINs. Because HINs are mainly constructed by information retrieving techniques nowadays, the unreliability of web sources and the biases of techniques [2] may lead to subtle errors. Downstream analysis tasks employing these biased data will cause accumulated errors in the applications. Therefore, the task of evaluating the

trustworthiness of HINs should be stressed to maintain the reliability of HIN data and improve the accuracy of downstream tasks [3]. Specifically, let  $t := (N_s, E_p, N_o)_G$  be a link in the HIN  $G$ , starting from the subject node  $N_s$  and ending with the objective node  $N_o$  with the corresponding edge  $E_p$ . Thus, worthiness evaluation tasks aim at computing a probabilistic trustworthiness  $T(w_G^t) \in [0, 1]$  for every link  $t$  in  $G$ . Here,  $T(w_G^t) = 1$  indicates link  $t$  is completely correct while  $T(w_G^t) = 0$  when  $t$  is totally incorrect.

Current HIN trustworthiness evaluation methods are mainly based on knowledge base (KB) methods [4]. They focus on inferring a link's trustworthiness in a single HIN, where the trustworthiness metric should evaluate how reliable this link is. In addition to basic characteristics like node types [5], attributed values [6], and confidence of extracted source, trustworthiness is typically evaluated from other perspectives. Network embedding methods [7] consider whether nodes and edges satisfy in the network

topology. For instance, R-GCN models relational data in the GCN framework. On the other hand, edge learning methods [3, 8–10] measure the acceptability of a link composed of an edge and two nodes corresponding to it. For example, TransE [9] considers whether the embedding nodes and edges of multi-relational data converge in low-dimensional vector spaces.

Almost all the previous trustworthiness evaluation methods are focusing on only one HIN; however, the information in a single HIN tends to be uncompleted or one-sided. Notice that multiple HINs may capture information from the same resources, and it is crucial to accomplish trustworthiness evaluation with the help of multiple HINs that model the same or similar domains which is shown in Figure 1. For example, HIN 1 has a description of Leonardo DiCaprio starring in *The Great Gatsby* where no connection between actors are shown in this HIN, while HIN 2 has links connecting actors, for example, Leonardo DiCaprio has cooperated with Matthew McConaughey. Thus, we can infer more information about Leonardo DiCaprio’s starred films. Different HINs usually have different emphasis even on the same domain. For example, two HINs constructed based on IMDB and YouTube, respectively, are different in many aspects even when they all model the information of videos. In one HIN, some links have ample evidence, while related links in another HIN only have little evidence such that it is difficult to evaluate trustworthiness in single HIN. Thus, in the multi-HIN situation, trustworthiness evaluation can be different as links in one HIN can influence their counterparts in another one. These related links from different HINs are called interactive links, and shared information of interactive links should improve the effectiveness of evaluation.

However, evaluating trustworthiness with multiple HINs is a nontrivial task because of several challenges:

- (1) *Aligning Interactivity.* The first challenge is how to determine the interactivity. If interactive links are only based on the same nodes or edges, the influence scope will be restricted, and the error introduced by the alignment procedure will deduce the evaluation result.
- (2) *Evaluating Influence across Multiple HINs.* The second challenge is to design an influential method across HINs. Having determined the interactive degree between a pair of interactive links, how to update this pair should be carefully processed.
- (3) *Balancing Inner-HIN and Inter-HIN Trustworthiness.* It is difficult to balance the inner-HIN and inter-HIN trustworthiness. It will be redundant and laborious to initially evaluate a link’s trustworthiness in every single graph and then evaluate it again in the co-embedding space.

These three difficulties make multi-HIN trustworthiness problems different from and much more difficult than traditional single HIN settings.

To address challenges in multi-HIN trustworthiness evaluation tasks, we propose a multi-HIN trustworthiness evaluation (multi-HITE) model, aiming at leveraging inner-

HIN characteristics and inter-HIN information in the co-embedding space to evaluate link trustworthiness. Three aforementioned challenges of multi-HIN tasks are solved accordingly through the following:

- (1) *Multiple Interactive Links Effect Functions.* We design three kinds of functions to evaluate the interactive links effect (ILE) across different HINs. These ILE functions evaluate the interactivity from literal level, semantic level, and graph level, respectively. Altogether, these functions help our model align interactivity across multiple HINs.
- (2) *Iterative Procedure to Update Trustworthiness.* Influential information across HINs is updated iteratively in the training process through threshold-based influence function.
- (3) *Integrated Learning with Multi-HINs.* The loss function of our model measures the inner-HIN and inter-HIN trustworthiness of multiple HINs at the same time. Additionally, the trade-off between these two parts is controlled by hyperparameters.

Specifically, this paper has made the following contributions:

- (1) To our best knowledge, our method is the first one considering evaluating the trustworthiness of HIN links in multi-HIN settings.
- (2) We propose a novel multi-HIN trustworthiness evaluation method combining inner-HIN and inter-HIN characteristics in a co-embedding space.
- (3) We conduct experiments on a real-world dataset that proves the effectiveness of our model. Based on different kinds of interactive metrics, we conduct several experiments using a real-world dataset compared with modified single-HIN evaluation methods on the tasks of error detection and link prediction.

*Notations.* A HIN is denoted by  $G$  and  $t = (N_s, E_p, N_o)_G$  represents one link in  $G$ . A set of multiple HINs is represented as  $S = \{G_1, G_2, \dots, G_K\}$ . Let  $N_s$ ,  $N_o$ , and  $E_p$  denote subjective node, objective node, and node relations in the original spaces and let  $N_s$ ,  $N_o$ , and  $E_p$ , respectively, denote their representations in the embedding space. Given a vector  $x \in \mathbb{R}^p$ ,  $\|x\| = \sqrt{x_1^2 + x_2^2 + \dots + x_p^2}$  denotes its  $l_2$ -norm, and given a matrix  $X \in \mathbb{R}^{m \times n}$ , its  $l_2$ -norm is defined in an elementwise manner as  $\|X\| = \sqrt{\sum_{1 \leq i \leq m, 1 \leq j \leq n} (X_{ij})^2}$ .

## 2. Materials and Methods

*2.1. Trustworthiness in a Single Network.* Due to the concerns about the influence of data quality on downstream tasks, link trustworthiness evaluation in an HIN and of a HIN source has quickly gained popularity among researchers [11]. Most existing methods are based on knowledge bases. Early methods are rule-based or statistical. For example, Ma et al. [12] learned disjointness axiom via association rule mining

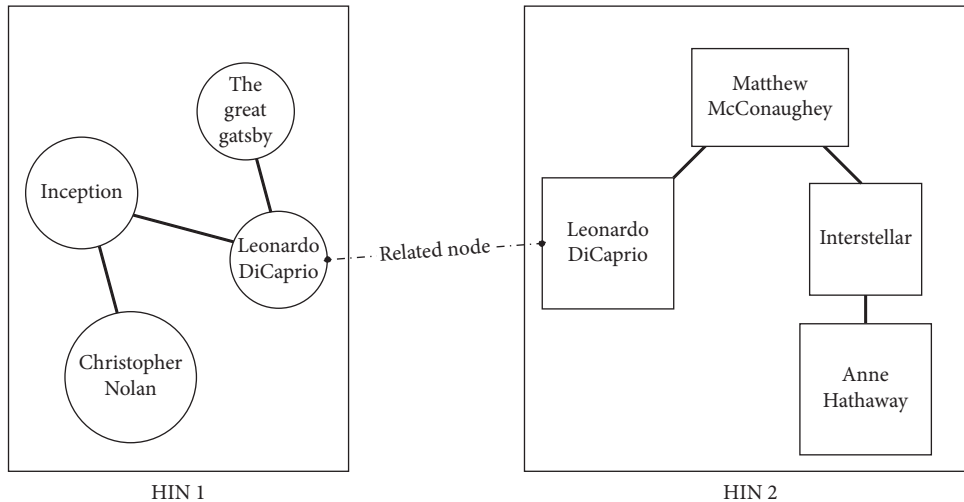


FIGURE 1: General situation of multi-HIN problems.

to dynamically update rule patterns. A statistical example is SDValidate [5], which detects noise in a HIN via the statistical distribution of the properties and types.

Another popular approach is to integrate the KB representation learning into the models to evaluate the trustworthiness in embedding space. Typical examples are the translation models, including TransE [9] and its variants like TransH [8] and TransR [13]. In Trans-series model, if the energy function of a link in the embedding space

$$A(w_G^t) = \|N_s + E_p - N_o\| \quad (1)$$

scores larger than a desired value, it is assumed that this link has little local context proof in original HIN space, i.e., having low trustworthiness.

In addition to translation models, CKRL [11] exploits the structure information by defining local link confidences and global path confidence to detect noises in HINs. This method exploits the structure information of the network. KGTtm [10] is a multi-level knowledge graph link trustworthiness evaluation method. It separates evaluation into nodes, edges, and graph levels, combining the internal semantic information of links and the global inference information of the KB. However, none of these methods consider utilizing other HINs that model similar domains. Therefore, their performance is limited as the information from a single HIN tends to be incomplete or one-sided.

**2.2. Learning across Multiple HINs.** In the multi-HIN setting, as nodes and edges in different HINs are extracted from different sources, network alignment methods are proposed to find corresponding nodes across multiple networks. Typically, network alignment can be categorized into local alignment and global alignment. Local alignment aims to find similar local regions across networks [14]. Global alignment focuses on exploiting the global topological consistency, such as in iNEAT [15] and CALign [16].

Conventional KB methods briefly use handcrafted “sameAs” link, literal information, and attribute features to

generate linkages between different HINs [17, 18]. Recent works focused on representation learning across multi-relational graphs, noticing that aligned nodes should be similar in space [19]. MTransE [20] tries to merge graphs and uses relational learning methods to discover aligned nodes. It combines three different transformation methods to learn embedding for supporting multi-lingual learning. LinkNbed [21] learns latent representations of nodes and edges in multiple graphs where a unified graph embedding is constructed, avoiding the bias caused by the transformation between different HIN embeddings.

We naturally consider utilizing similar methods in node alignment in the multi-HIN trustworthiness evaluation problem. However, these methods of the node alignment problem lack global structure inference information, which prevents these methods from being directly generalized to multi-HIN trustworthiness evaluation.

**2.3. General Structure of Trustworthiness Evaluation.** To address the limitations of previous evaluation methods, we propose a novel method inherently considering inner-HIN and inter-HIN effects. The trustworthiness evaluation in multi-HIN settings can be separated into two parts. The general idea is shown in Figure 2.

Firstly, given the overall trustworthiness  $V_{G_i}$  of the  $i$ -th HIN, evaluate trustworthiness score of all links  $t_j \in G_i$  because in the multi-HIN situation, different HINs have different reliabilities. The trustworthiness of a link in graph  $G_i$  is not only based on the inner-HIN trustworthiness  $A(w_{G_i}^{t_j})$  calculated by inner characteristics of  $G_i$  but also dependent on the HIN source confidence  $V_{G_i}$ . This influence can be denoted as

$$T(w_{G_i}^{t_j}) = V_{G_i} A(w_{G_i}^{t_j}). \quad (2)$$

Initially, all global confidence  $V_{G_i}$  is assigned to 1. Thus, according to the representations for head nodes, tail nodes, and edges which are generated by inner characteristics and initial source trustworthiness, the trustworthiness of links can be calculated for every HIN.

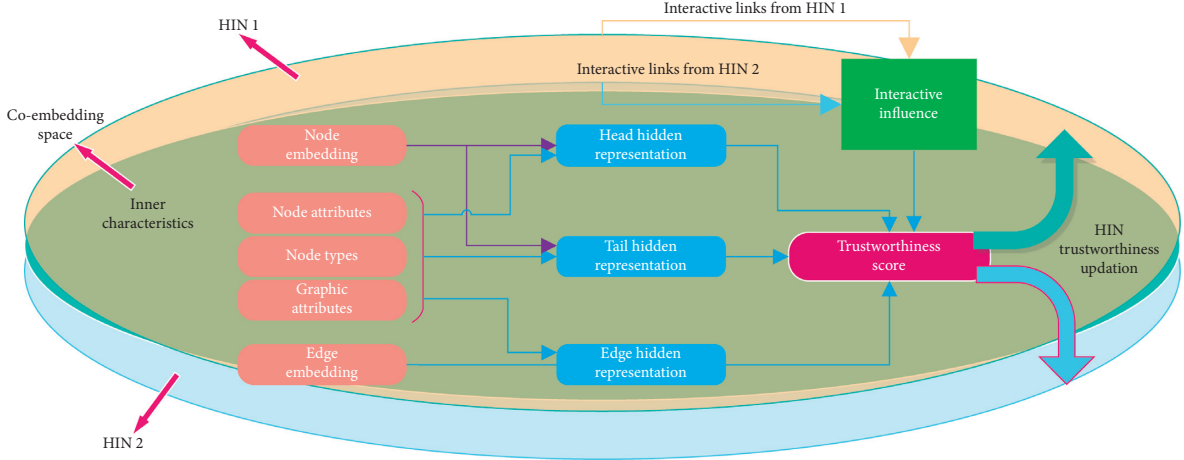


FIGURE 2: General structure of multi-HIN trustworthiness evaluation method.

Then, in the second step, we use link trustworthiness score to conversely update  $V_{G_i}$  based on  $T(w_{G_i}^{t_j})$ . In the opposite view of the multi-HIN problem, we assume that the trustworthiness of a HIN source is determined by all links' trustworthiness in the source and is computed via

$$V_{G_i} = \frac{1}{n_{G_i}} \sum_{t_j \in G_i} T(w_{G_i}^{t_j}). \quad (3)$$

Here,  $n_{G_i}$  means the total number of links in graph  $G_i$ , which acts as a normalization factor. At each iteration,  $V_{G_i}$  is firstly fixed to calculate single link's trustworthiness according to equation (2) followed by fixing  $T(w_{G_i}^{t_j})$  to calculate  $V_{G_i}$  according to equation (3). After the iterative training has converged, a steady trustworthiness evaluation of HIN source and link can be obtained.

The above procedure is commonly used in no-mutual-influence multi-HIN settings. As for mutual-influence multi-HIN cases, the influence between different HINs needs to be modeled, where our method models this influence by the trustworthiness influences between interactive links across HINs. This will be discussed in "Inter-HIN Evaluation" Section. Specifically, different HINs are placed on the co-embedding space to represent the influence between the sources. If HINs are separately evaluated in their own space, this will cause redundant procedure in the transformation between HIN spaces which make the model difficult to train. Then, having obtained inner scores for links in both HINs, interactive influence will update corresponding links' scores. The final step is to update the source trustworthiness. This process will be iteratively conducted. All HIN sources' trustworthiness values are initially set to 1 and after iterative influencing between sources and links, the final scores of links and sources will be obtained.

Because the process of trustworthiness evaluation is the same in all the HINs, without loss of generality, we use  $G$  and  $t$  to denote an arbitrary HIN and a certain edge in it. Without explicit statements, we drop the subscripts  $i$  and  $j$  for simplicity in the rest of this paper.

The following sections will be organized as follows: we detail our model's trustworthiness evaluation in a single HIN in Subsection 2.4, the details of the multi-HIN mutual

influence method in Subsection 2.5, and the training method tailed for multi-HIN evaluation in Subsection 2.6.

**2.4. Inner-HIN Evaluation.** We start by considering how to compute the link trustworthiness score with inner-HIN characteristics. Based on the current single HIN trustworthiness evaluation methods [22], HIN's inner characteristics, including semantic and graphic information, are strong proofs to evaluate the reliability of links. In our model, we utilize four inner-HIN characteristics including node attributes, node-type constraints, contextual embeddings, and graphic energy transmission information to evaluate links.

**2.4.1. Node-Attribute Information.** Node attributes are the basic and effective information to assess whether a node is consistent with its value. Here we use doc2vec [23] to embed attribute values and then integrate the value with its one-hot attribute key index to represent hidden information of nodes. If a node has several attributes, the hidden attribute information will be averaged.

**2.4.2. Node-Type Constraint.** For a certain edge, the nodes appearing near it normally have prevalent and uniform types. Thus, for each edge, we can summarize the node types that emerge around it. Then, the node-type information and edge embedding information are integrated into the hidden layer.

**2.4.3. Contextual Embedding Information.** Trans-series models embed nodes and edges in low dimension representation space. These embeddings contain global contextual information where outliers can be found based on the inconsistency. However, we are not directly using the energy function to evaluate the consistency. Here, we utilize the fully connected layer to unite other inner-HIN characteristics with embeddings to get a hidden layer's representation.



**2.4.4. Graph Energy Transferring Information.** HIN is a special heterogeneous network, where basic network analysis techniques can be utilized. If a link has strong connectivity or other favourable graphic features, it also indicates sufficient proofs in the HIN. Here we mainly concentrate on the energy flow between the head node and the tail node. In this situation, we can construct a single graph for each node in the HIN, where the node is the central node. All nodes associated with the central node are taken into consideration. The energy transferring can be modeled with page-rank-like method [11, 24]. However, iterating through a graph will have a long tail phenomenon where the majority of nodes have low energy. Hence, we add several features like in-degree and out-degree into consideration as

$$g(N_s, N_o) = \varepsilon \times \frac{e(N_s, N_o)}{e(N_s)} + (1 - \varepsilon) \times f_{N_s, N_o}(d_{in}, d_{out}). \quad (4)$$

Here,  $g(N_s, N_o)$  is the combined energy transmitted between embedded nodes  $N_s$  and  $N_o$ .  $e(N_s)$  represents the energy that  $N_s$  holds in the graph, and  $e(N_s, N_o)$  denotes the energy flowing from  $N_s$  to  $N_o$ .  $f_{N_s, N_o}(d_{in}, d_{out})$  is the energy function determined by the in-degree and out-degree of  $N_s$  and  $N_o$ . Besides,  $\varepsilon$  is a hyperparameter balancing the page-rank-like energy and degree-determined function.

After obtaining the above four different features, each feature is transmitted to the hidden dimension by a separate linear forward layer followed by the combination of all features of the same node or link, as shown in Figure 2. After obtaining the hidden representation  $z \in \mathbb{R}^n$  ( $n$  is the size of hidden states) of the head node, tail node, and edge, we use the following function to model the trustworthiness:

$$T(w_G^t) = V_G \times \sigma(z_p^\top (z_s \odot z_o)), \quad (5)$$

in which  $\sigma(\cdot)$  is the activation function of the neural network and  $\odot$  denotes the elementwise multiplication. The score function is related to the score of the link's located HIN, so it needs to multiply the located HIN's score  $V_G$ . Part of this function is initially adopted by DistMult [25].

With all these, we can calculate traditional transmission loss in every single HIN of  $S$ . This loss is used to model local information of a HIN. For a specific link  $t$  shown in the original dataset, we call it a positive link. The negative links set  $c(G)$  is generated by replacing the head and tail node of  $t$  with other nodes so that negative links do not appear in the original dataset. Specifically, for a HIN  $G \in S$ , let  $T(w_G^t)$  and  $T(w_G^{t'})$  be the positive score and negative score, respectively, for the positive link  $t$  and negative link  $t'$  in this HIN, and the relational loss for multiple HINs is

$$L_{rel} = \sum_{G \in S} \left( \sum_{t \in G} \sum_{t' \in c(G)} \max(0, \gamma - T(w_G^t) + T(w_G^{t'})) \right). \quad (6)$$

Here,  $\gamma$  is the margin between positive links and negative links, and  $c(G)$  is a set of corrupted links  $t'$  corresponding to links  $t$  in the HIN  $G$ .

**2.5. Inter-HIN Evaluation.** In multi-HIN settings, inter-HIN influence is another crucial component of the evaluation. Therefore, we further include inter-HIN influence into the loss function for trustworthiness evaluation. If a link  $(N_s, E_p, N_o)_G$  in one HIN  $G$  has influence on a link  $(N_{s'}, E_{p'}, N_{o'})_{G'}$  in another HIN  $G'$ , we call these two links interactive links. The way how interactive links influence each other in different HINs is called *Interactive Links Effect (ILE)*. We use  $h(\cdot)$  to denote ILE and propose three different kinds of ILEs that are utilized in our method for modeling interactivity from three levels: literal level, semantic level, and graph level.

**2.5.1. Value ILE.** If two nodes refer to the same real-world node, they should share similar attributes [26]. Based on this assumption, from the literal level, the value ILE can be defined to evaluate influence for nodes of a pair of interactive links. Let  $p(N)$  denote the attribute value set of the node  $N$  and  $c(N_1, N_2)$  represent the common attributes shared by nodes  $N_1$  and  $N_2$  in the embedding space; value ILE is computed via

$$\begin{aligned} h_{\text{value}}\left(\left(N_s, E_p, N_o\right)_G, \left(N_{s'}, E_{p'}, N_{o'}\right)_{G'}\right) \\ = \frac{|c(N_s, N_{s'})|}{|p(N_s)| + |p(N_{s'})| - |c(N_s, N_{s'})|} \\ + \frac{|c(N_o, N_{o'})|}{|p(N_o)| + |p(N_{o'})| - |c(N_o, N_{o'})|} \end{aligned} \quad (7)$$

in which  $|\cdot|$  is the number of elements in a set.

**2.5.2. Alignment ILE.** If two links have aligned nodes or edges, they should be interactive. Additionally, aligned nodes or edges are close in embedding space [19]. Thus, we can assume that if two links are interactive, at least one of the nodes or edges should be close in the embedding space. Besides, only one aligned part is not sufficient as many links describe the same node from a different perspective like ‘‘Yao Ming-wife-Ye Li’’ and ‘‘Yao Ming- teammate-Tracy McGrady.’’ Thus, we add separate distances of co-space embedding of nodes and edges together and restrict the distance of interactive links of this embedding distance to determine whether two links are interactive or not. With this, we define the alignment ILE as

$$\begin{aligned} h_{\text{alignment}}\left(\left(N_s, E_p, N_o\right)_G, \left(N_{s'}, E_{p'}, N_{o'}\right)_{G'}\right) \\ = \|N_s - N_{s'}\|_2 + \|N_o - N_{o'}\|_2 + \|E_p - E_{p'}\|_2. \end{aligned} \quad (8)$$

**2.5.3. Neighbour ILE.** According to the semantic similarity work [27], if nodes share similar surroundings, there exists a possibility that they have semantic similarity; in other words, they are interactive. Thus, we use neighbouring nodes of the head and tail nodes to evaluate the ILE. Let  $n(N)$  denote the average embedding for neighbouring nodes of node  $N$ ; then, we have

$$h_{\text{neighbour}}\left(\left(N_s, E_p, N_o\right)_G, \left(N_{s'}, E_{p'}, N_{o'}\right)_{G'}\right) = \left\|n\left(N_s\right) - n\left(N_{s'}\right)\right\|_2 + \left\|n\left(N_o\right) - n\left(N_{o'}\right)\right\|_2. \quad (9)$$

With all three ILE functions, we can inherently use them to determine if two links from two different HINs have influence on each other. Specifically, we design a matrix  $I$  to represent the interaction of two HINs. For a certain link  $i \in G$  and another link  $j \in G'$ , their interactive status is defined as

$$I_{i,j} = 1\left(h_{\text{value}}\left(i, j\right) < \theta_{\text{value}}\right) + 1\left(h_{\text{alignment}}\left(i, j\right) < \theta_{\text{alignment}}\right) + 1\left(h_{\text{neighbour}}\left(i, j\right) < \theta_{\text{neighbour}}\right), \quad (10)$$

where  $1(c)$  is the indication function with  $1(c) = 1$  if statement  $c$  is true; otherwise,  $1(c) = 0$ . Three ILE thresholds including  $\theta_{\text{value}}$ ,  $\theta_{\text{alignment}}$ , and  $\theta_{\text{neighbour}}$  are playing the roles of hyperparameters.

Intuitively, the trustworthiness of two interactive links should be close to each other. However, there exist cases where two links are interactive, but one has large trustworthiness while that of another link is low. Thus, trustworthiness scores of two links should be deducted. Specifically, for  $i \in G$  and  $j \in G'$ , we design the influence between links  $i$  and  $j$  as  $R(i, j) = 0$  if  $I_{i,j} = 0$ ; otherwise,

$$R(i, j) = \begin{cases} \left|T\left(w_G^i\right) - T\left(w_G^j\right)\right|, & \text{if } \left|T\left(w_G^i\right) - T\left(w_G^j\right)\right| \leq \rho, \\ \left|T\left(w_G^i\right) + T\left(w_G^j\right)\right|, & \text{if } \left|T\left(w_G^i\right) - T\left(w_G^j\right)\right| > \rho. \end{cases} \quad (11)$$

Combining all the inter-HIN influences of interactive links, we introduce the interactive loss:

$$L_{\text{int}} = \sum_{(i,j) \in \text{In}} R(i, j), \quad (12)$$

with a set of interactive links  $\text{In}$ .

**2.6. Model Training in Multi-HIN Settings.** To obtain the trustworthiness both in single-HIN and multi-HIN perspectives, we introduce the multi-task objective function, which models the contextual information of local and influential information. We integrate relational loss with interactive influential loss by using a hyperparameter  $\beta$  as the trade-off and obtain

$$L = \beta L_{\text{rel}} + (1 - \beta)L_{\text{int}}. \quad (13)$$

The training process of our model can be separated into two main procedures.

In the first phase, we minimize the loss function with a neural network. We put inner graph characteristics into the neural network, including node attributes, node-type constraints, node and edge embeddings, and graph energy transferring information. Then, we separately combine node-related features into hidden representations of nodes and combine node-type constraints with edge embeddings into hidden representations of edges. The detailed structure

of this neural network is shown in Figure 3. In the network,  $e_1(\cdot)$  is the node energy function for a certain node.  $e_2(\cdot)$  is the node embedding,  $e_3(\cdot)$  denotes the attribute embedding,  $e_4(\cdot)$  represents the edge embedding, and  $e_5(\cdot)$  is the type embedding of a node. Moreover, weight matrices  $W_{e1h}$ ,  $W_{e2h}$ ,  $W_{e3h}$ ,  $W_{e4h}$ , and  $W_{e5h}$  project different features to its hidden representations. Function  $r(\cdot)$  projects embeddings in the same space. Besides,  $W_{er}$ ,  $W_{nr}$ , and  $W_{ar}$  combine node-related features to generate hidden representations of nodes while  $W_{lr}$  and  $W_{tr}$  combine edge-related features to generate hidden representations of edges. In addition, we further attach an extra regularizer to control the size of all the parameters. Let  $\Omega = \{W_{e1h}, W_{e2h}, W_{e3h}, W_{e4h}, W_{e5h}, W_{er}, W_{nr}, W_{ar}, W_{lr}, W_{tr}\}$  denote a set of all the weight parameters in the neural network, and the regularized loss function becomes

$$L = \beta L_{\text{rel}} + (1 - \beta)L_{\text{int}} + \sum_{W \in \Omega} \|W\|_2. \quad (14)$$

After evaluating the score for all links in multiple HINs, we can iteratively use equations (2) and (3) to update the trustworthiness of HIN and each link, until results converge. The pseudocode of our method is shown in Algorithm 1.

### 3. Results and Discussion

In this section, we implement comprehensive experiments to test the performance of our model from various perspectives.

**3.1. Dataset.** We choose to use two HIN alignment datasets **DBP-WD** and **DBP-YG** [28] in our experiments. The first dataset is sampled from DBpedia and Wikidata (i.e., two HINs) and contains 100 thousand aligned nodes, while the second dataset is extracted from DBpedia and YAGO with same 100 thousand aligned nodes. Information about node types is attached to the data according to DBpedia type information. Tables 1 and 2 provide statistics of these datasets in the experiment. We split this dataset into training, testing, and validation set with the ratio as 10:1:1. Besides, DBP-WD dataset is used in all experiments while DBP-YG is only used in Experiment III. DBpedia in DBP-WD is noted as DBpedia while DBpedia in DBP-YG is noted as DBpedia2.

Furthermore, we generate some negative links in the dataset to simulate noisy data with erroneous links. Specifically, for one negative link, we replace the head or tail node of a positive link so that it does not appear in the original dataset. Other characteristics including node attributes, node types, and neighbouring nodes remain the same with the replaced node. To obtain a different level of noise, similar to the simulation implemented in [11], multiple noisy datasets are simulated with the percentage of replaced negative links of 10%, 20%, and 40%. Notice that the noises are only added to the training set.

**3.2. Baseline Methods.** As there is no trustworthiness evaluation method previously proposed in multi-HIN settings, we directly extend traditional embedding methods into multi-HIN cases as baseline methods. We use embedding

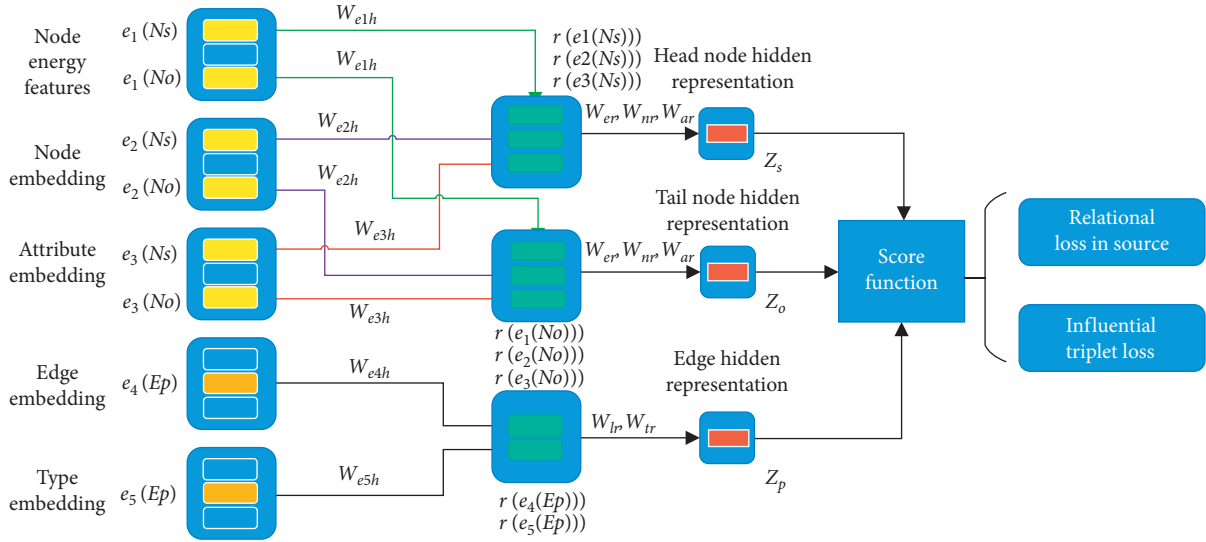


FIGURE 3: Detailed structure of neural network.

```

REQUIRE:  $S$ , maximal epochs of training  $T$ ; source trustworthiness  $V$ .
for  $G \in S$  do
   $V_G \leftarrow 1$ 
end for
while  $T > 0$  do
  for  $G \in S$  do
    for  $t \in G$  do
      Calculate link score according to (2)
    end for
  end for
  for  $G \in S$  do
    Update source trustworthiness  $V_G$ 
  end for
   $T \leftarrow T - 1$ 
  Backpropagate loss and update parameters in the neural network
end while

```

ALGORITHM 1: Multi-HIN trustworthiness evaluation.

TABLE 1: Statistics of HINs constructed based on DBpedia and Wikidata.

HIN	# of nodes	# of edges	# of attributes	# of node types	# of links
DBpedia1	100000	686	622331	225	463294
Wikidata	100000	946	990517	295	448774

methods including **TransE** [9], **R-GCN** [29], **DistMult** [25], and **ComplexE** [30]. However, we are not using original versions of these algorithms. Instead, we use refined models from **OpenKE** [31] to compare our model with current frequently used model versions. Because these methods do not generate trustworthiness scores directly, we calculate the trustworthiness score according to their threshold where a link score that is larger than the threshold is thought as 0, otherwise as 1. Moreover, because these methods are not proposed for multi-HIN cases, they are not able to integrally

analyse multiple HINs. Thus, we use these methods to evaluate trustworthiness scores separately on different HINs.

Additionally, we simplify our model for comparisons between our model with its simplified version. We use two simplified models to show the effectiveness of our model setting. These two versions of methods are denoted as  $M_{\text{attribute-}}$  and  $M_{\text{influence-}}$  respectively.  $M_{\text{attribute-}}$  is a version ignoring node attribute inner-HIN characteristic where in the inner-HIN evaluation stage, only node-type constraint, contextual embedding information, and graph energy

transferring information are taken into consideration.  $M_{\text{influence-}}$  is the simplified model neglecting influence methods through ILE between HINs, where in the inter-HIN evaluation stage, all ILEs are set to 0 so that there exist no interactive links.

**3.3. Hyperparameter Setup.** In the neural network of our model, the size of the node embedding is 256, the edge embedding size is 64, and the attribute embedding size is 512. The hidden size and combined node/edge size is 512. We use the ReLU function as the activation function in the model. The sigmoid function is used in the output layer for producing the trustworthiness score. The value of learning rate is selected from  $\{0.1, 0.01, 0.001\}$  and set as 0.001, and the batch size is selected from  $\{10, 40, 100\}$  and set as 0.001. These hyperparameters are selected through cross-validation with the best performance in multi-HIN situations.

The value of margin  $\gamma$  in our function is selected from  $\{0.5, 0.8, 1\}$ . Moreover, we use  $l_2$ -norm as the regularizer to control all the trainable parameters in the neural network. Inner-HIN loss and cross-HIN balance parameter  $\beta$  is selected from  $\{0.3, 0.5, 0.7\}$  and set as 0.5. Graphic balancing parameter  $\epsilon$  is selected from  $\{0.3, 0.5, 0.7\}$  and set as 0.5. Influence method control threshold parameter  $\rho$  is selected from  $\{0.01, 0.1, 0.5\}$  and set as 0.1. Value influence threshold  $\theta_{\text{value}}$  is selected from  $\{0.5, 0.8, 0.9\}$  and set as 0.8. Alignment influence threshold  $\theta_{\text{alignment}}$  is selected from  $\{0, 0.1, 0.2, 0.4\}$  and set as 0.2. Neighbouring influence threshold  $\theta_{\text{neighbor}}$  is selected from  $\{0, 0.1, 0.2, 0.4\}$  and set as 0.1. These hyperparameters are selected through cross-validation with the best performance in multi-HIN situations.

**3.4. Experiment I: Errorless Link Classification.** Link classification is the task of testing the model trustworthiness evaluation ability in new data. In our experiment, this experiment is implemented on the testing set. It is a simplified task compared with error link classification and link error detection [11] where there is no noise in the training set. The correct link is labelled as 1 while a false link is labelled as 0. The model output lies in the range of  $[0, 1]$ , representing the probability of one link being a correct one. The trustworthiness score is binarized into 0 (false link label) or 1 (correct link label) based on a threshold value. Namely, given a link, the trustworthiness score is calculated according to equation (5). If the trustworthiness score is larger than the threshold, this link is labelled as the correct link, otherwise, the false link.

Here, we compare our model with other models in basic situations, namely, in separate HIN space and in combined HIN spaces. Besides, in this experiment, we train and test on the original dataset, not adding false links into training. Also, assuming we do not know where the link comes from, the HIN source trustworthiness  $V_G$  is set to 1.

Experimental results are shown in Table 3. Obviously, our model outperforms baseline methods in separate HIN and combined HIN space, which indicates that the hidden information representing the reliability of links has been well learned in our model. We can analyse this result from two different perspectives: (1) single HIN and multi-HIN; (2)

influence of inner-HIN characteristics. From the view of single and multi-HIN, we can notice that when HINs are combined, in other words, when data are augmented, evaluation results are improving compared with those in single HIN situation. This can be also proved from the threshold dropping in TransE, DistMult, and partial ComplexE results. This proves that though previous methods do not directly consider the connection between HINs, they can model the inherent relationship between multiple HINs in other implicit ways while their results are no better than our explicit relationship modeling method. From the view of influence from inner-HIN characteristics, attribute characteristics are rather important in the link classification task, as we can see the accuracy of  $M_{\text{attribute-}}$  drops by nearly 5% compared with normal multi-HITE model. Our model can achieve convergence by running two or three epochs on the training data. In the DBP-WD dataset, it takes 45 minutes for one epoch. The network embedding model usually takes two hours to network embedding, followed by the training of the classification task which costs more time than ours.

**3.5. Experiment II: Error Link Classification and Link Error Detection in Two HINs.** The evaluation metric is the same as the error detection task. The first task, error link classification, is similar to the errorless link classification. The difference is in the training set, where error link classification is trained on the noise training set while testing set and validation set are the same. A small change in our model takes place here, as our model cannot converge optimal in the noise training set, which means our model detects noise. We first train and evaluate our model on the training set (both positive and negative-sampled links) and then retrain our model on the generated training set where labels are predicted from the first trained model’s evaluation. The second task, link error detection, is conducted on the training set. The results are shown in Figure 4 in the format of PR curve to test different models’ performance on the binary classification task of detecting error links.

In Table 4, we can see our model achieves the best accuracy and  $F1$ -score, when compared with other models in the multi-HIN setting for error link classification in 10% and 40% neg situation. In detail, our model thought 786856, 816749, and 895114 links may be correct in the first train and evaluation round separately (where 757126, 757135, and 757139 are truly correct), where 10%, 20%, and 40% noise datasets contain links of 834973, 910880, and 1062693. When the noise links rise, our model is more likely to rule out false examples. It reveals that our model achieves effectiveness in modeling the inter-influence of HINs by using interactive links. For TransE, DistMult, and ComplexE, they are robust to errors in the training set where errors slightly reduce models’ performance. It reveals that the traditional embedding model can handle the problem of error links in multi-HIN trustworthiness situations. Two simplified models indicate the importance of inner characteristics, and we can conclude that the attributes are an important indicator of whether a link is true or not. We think that the reason why attribute factors play a more important role in



TABLE 2: Statistics of HINs constructed based on DBpedia and YAGO.

HIN	# of nodes	# of edges	# of attributes	# of node types	# of links
DBpedia2	100000	643	760062	225	428952
YAGO	100000	64	827682	295	502563

TABLE 3: Performance of link prediction from all the models on multiple HINs.

Data	Multi-HITE		$M_{\text{influence-}}$		$M_{\text{attribute-}}$		TransE		R-GCN		DistMult		ComplexE	
	acc.	F1	acc.	F1	acc.	F1	acc.	F1	acc.	F1	acc.	F1	acc.	F1
DBpedia1	—	—	0.92	0.92	0.87	0.88	0.91	0.91	0.88	0.88	0.90	0.89	0.91	0.90
Wikidata	—	—	0.94	0.94	0.93	0.89	0.90	0.90	0.89	0.89	0.88	0.88	0.89	0.88
Combined	0.95	0.95	0.93	0.93	0.90	0.90	0.92	0.92	0.90	0.90	0.91	0.91	0.91	0.91

“acc.” and “F1” denote the accuracy and F1-score of estimations, respectively.

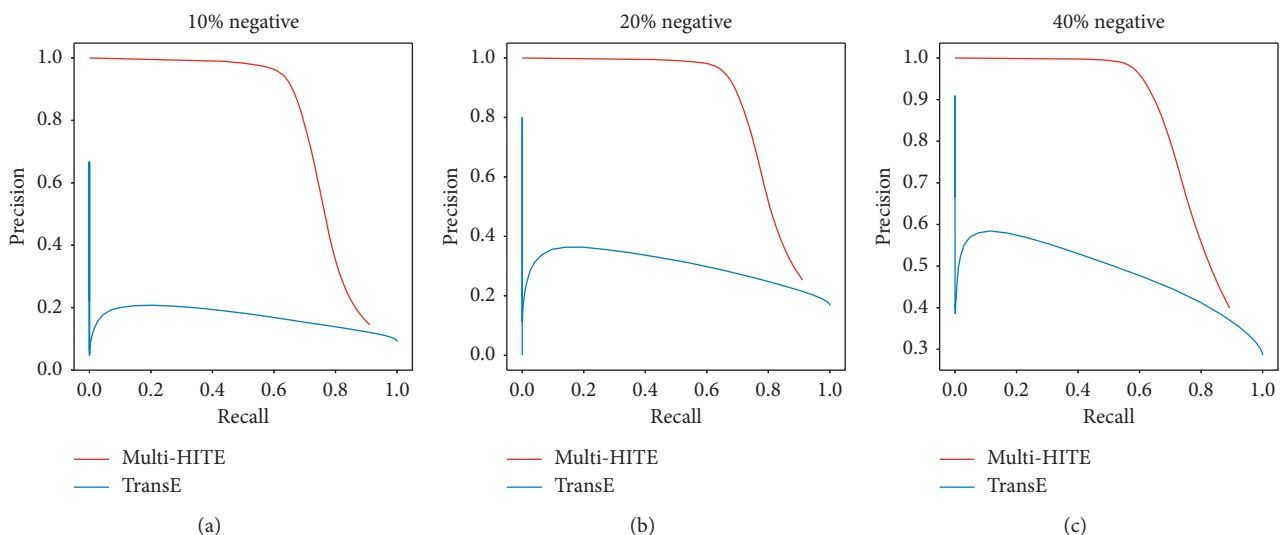


FIGURE 4: Trade-off between precision and recall according to trustworthiness threshold.

the evaluation is the number of attributes. For 200,000 entities in this paper’s dataset, there are 1,612,848 attribute descriptions in total, while the interactions between HINs are subject to mutual influence conditions, where the number of influence restrictions is far smaller than that of attribute description information. Therefore, deleting attribute information will affect the performance of the model more than deleting interaction information.

In Figure 4, we can see our model’s PR curve is above TransE’s PR curves in the multi-HIN setting for link error detection in all neg situation. DistMult and ComplexE cannot distinguish noise links in the training set, so they are not plotted in Figure 4. It reveals that our model achieves effectiveness in detecting noise in multi-HIN situation. Figure 4 also explains that there is a difference between the distributions of true links and noise links’ scores both in multi-HITE and TransE model. For lines of TransE in Figure 4, there exist sharp drops in the left part. This can be explained that the scores for true links and noisy links do not have a large gap, especially near extreme threshold where all links are thought as positive. When the threshold drops, few links are thought to be noise while positive links are mistaken as noise, which will influence the curve

sharply. Also, it can be observed that the PR curve rises when the number of noise links increases as it is easier to detect errors when errors are more frequent.

**3.6. Experiment III: Error Link Classification in Different Combinations of HINs.** In this part, we try to combine different HINs with different numbers of HINs to compare the performance of our model for error link classification. The results are shown in Table 5. In the case of combining different data sources, our model proves robust from the results. At the same time, with the increment of data sources, our model can make more use of the interaction information between different data sources to provide more sufficient evidence for link classification.

**3.7. Experiment IV: Influential Method Evaluation.** Finally, we test how the performance of two ILEs varies with different values of the threshold and test the model in errorless link classification tasks. Based on the results in Table 6, we can conclude that alignment ILE is more sensitive to the threshold value because its threshold is close to 0

TABLE 4: Performance of all the methods over data with different levels of noise.

Data	Multi-HITE		$M_{\text{influence-}}$		$M_{\text{attribute-}}$		TransE		R-GCN		DistMult		ComplexE	
	acc.	F1	acc.	F1	acc.	F1	acc.	F1	acc.	F1	acc.	F1	acc.	F1
10% neg.	0.91	0.92	0.90	0.90	0.90	0.89	0.91	0.90	0.90	0.90	0.91	0.91	0.91	0.91
20% neg.	0.90	0.91	0.89	0.89	0.89	0.89	0.91	0.90	0.89	0.89	0.90	0.90	0.90	0.90
40% neg.	0.92	0.92	0.90	0.89	0.89	0.89	0.90	0.90	0.89	0.89	0.88	0.88	0.88	0.88

“acc.” and “F1” denote the accuracy and F1-score of estimations, respectively.

TABLE 5: Performance of multi-HITE over different datasets with different levels of noise.

Dataset	DBP1-WD		DBP2-YAGO		YAGO-WD		DBP1-YAGO-DBP2-WD	
	acc.	F1	acc.	F1	acc.	F1	acc.	F1
10% neg.	0.91	0.92	0.90	0.90	0.91	0.91	0.91	0.92
20% neg.	0.90	0.91	0.90	0.90	0.89	0.89	0.91	0.92
40% neg.	0.92	0.92	0.89	0.89	0.89	0.89	0.90	0.90

“acc.” and “F1” denote the accuracy and F1-score of estimations, respectively.

TABLE 6: Results (accuracy/F1-score) of multi-TW using different values of thresholds for ILE functions.

Alignment neighbour	0.0	0.1	0.2	0.4
0.0	0.8995/0.8981	0.8994/0.9020	0.8968/0.8997	0.8433/0.8495
0.1	0.8723/0.8735	0.8937/0.8975	0.9058/0.9027	0.8883/0.8904
0.2	0.8639/0.8673	0.9480/0.9482	0.9233/0.9253	0.9104/0.9078
0.4	0.8438/0.8453	0.9225/0.9237	0.9404/0.9395	0.9075/0.9017

compared with neighbour ILE. It should be explained that neighbour ILE (see equation (9)) has fewer components compared with alignment ILE (see equation (8)).

## 4. Conclusions

In this paper, we focus on the multi-HIN trustworthiness evaluation problem and propose a co-embedding space evaluation method. Our model incorporates inner-HIN and inter-HIN characteristics into the loss function. A deep neural network is used in our model to solve the problem. We compare our model with other single-HIN trustworthiness evaluation methods on a multi-HIN dataset. Experimental results demonstrate that our model can well accomplish the trustworthiness evaluation task and outperforms baseline models. For future works, we will focus on how to incorporate truth value discovery into our model. Besides, the processing of our model is complicated compared with other models such as TransE, DistMult, and ComplexE. Therefore, we need to find a concise way to evaluate trustworthiness while maintaining the current level of accuracy.

## Data Availability

The HIN alignment dataset DBP-WD used to support the findings of this study can be accessed through <https://github.com/nju-websoft/BootEA>.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This study was supported by the National Key Research and Development Program of China (grant nos. 2018YFC0830201 and 2017YFB1002801), the Fundamental Research Funds for the Central Universities (grant nos. 4009009106 and 22120200184), and the CCF-Baidu Open Fund.

## References

- [1] F. M. Harper and J. A. Konstan, “The movielens datasets: history and context,” *ACM Transactions on Interactive Intelligent Systems*, vol. 5, no. 4, 2015.
- [2] H. Paulheim, “Knowledge graph refinement: a survey of approaches and evaluation methods,” *Semantic Web*, vol. 8, no. 3, pp. 489–508, 2017.
- [3] Y. Lin, Z. Liu, H. Luan et al., “Modeling relation paths for representation learning of knowledge bases,” in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 705–714, Lisbon, Portugal, September 2015.
- [4] C. Yang, Y. Xiao, Y. Zhang et al., “Heterogeneous network representation learning: survey, benchmark, evaluation, and beyond,” 2020, <http://arxiv.org/abs/2004.00216>.

- [5] H. Paulheim and C. Bizer, "Improving the quality of linked data using statistical distributions," *International Journal on Semantic Web and Information Systems*, vol. 10, no. 2, pp. 63–86, 2014.
- [6] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection," *ACM Computing Surveys*, vol. 41, no. 3, pp. 1–58, 2009.
- [7] Y. Dong, N. V. Chawla, and A. Swami, "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, Halifax, Canada, August 2017.
- [8] Z. Wang, J. Zhang, J. Feng et al., "Knowledge graph embedding by translating on hyperplanes," *AAAI*, vol. 14, pp. 1112–1119, 2014.
- [9] A. Bordes, N. Usunier, A. Garcia-Durán et al., "Translating embeddings for modeling multi-relational data," in *Proceedings of the 26th International Conference on Neural Information Processing Systems-Volume 2*, pp. 2787–2795, New York, NY, USA, December 2013.
- [10] S. Jia, Y. Xiang, X. Chen et al., "Triple trustworthiness measurement for knowledge graph," in *Proceedings of the World Wide Web Conference*, pp. 2865–2871, San Francisco, CA, USA, May 2019.
- [11] R. Xie, Z. Liu, F. Lin et al., "Does william shakespeare really write hamlet? Knowledge representation learning with confidence," in *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, pp. 4954–4961, New Orleans, LA, USA, February 2018.
- [12] Y. Ma, H. Gao, T. Wu et al., "Learning disjointness axioms with association rule mining and its application to inconsistency detection of linked data," in *Proceedings of Chinese Semantic Web and Web Science Conference*, pp. 29–41, Wuhan, China, August 2014.
- [13] H. Lin, Y. Liu, W. Wang, Y. Yue, and Z. Lin, "Learning entity and relation embeddings for knowledge resolution," *Procedia Computer Science*, vol. 108, pp. 345–354, 2017.
- [14] S. Zhang, H. Tong, R. Maciejewski et al., "Multilevel network alignment," in *Proceedings of the World Wide Web Conference*, pp. 2344–2354, San Francisco, CA, USA, May 2019.
- [15] S. Zhang, H. Tong, J. Tang et al., "Ineat: incomplete network alignment," in *Proceedings of the 2017 IEEE International Conference on Data Mining (ICDM)*, pp. 1189–1194, New Orleans, LA, USA, November, 2017.
- [16] K. Chao, "Calign: Aligning sequences with restricted affine gap penalties," *Bioinformatics*, vol. 15, no. 4, pp. 298–304, 1999.
- [17] S. Auer, C. Bizer, G. Kobilarov et al., "Dbpedia: a nucleus for a web of open data," in *Proceedings of the 6th International Semantic Web Conference*, pp. 722–735, Busan, Korea, November 2007.
- [18] J. Hoffart, F. M. Suchanek, K. Berberich, and G. Weikum, "Yago2: a spatially and temporally enhanced knowledge base from wikipedia," *Artificial Intelligence*, vol. 194, pp. 28–61, 2013.
- [19] L. Y. Wu, A. Fisch, S. Chopra et al., "Starspace: embed all the things!" in *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, pp. 5569–5577, New Orleans, LA, USA, February 2018.
- [20] M. Chen, Y. Tian, M. Yang et al., "Multilingual knowledge graph embeddings for cross-lingual knowledge alignment," in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 1511–1517, Melbourne, Australia, August 2017.
- [21] R. Trivedi, B. Sisman, X. L. Dong et al., "Linkbed: multi-graph representation learning with entity linkage," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 252–262, Melbourne, Australia, July 2018.
- [22] E. Huaman, E. Kärle, and D. Fensel, "Knowledge graph validation," 2020, <http://arxiv.org/abs/2005.01389>.
- [23] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the 31st International Conference on International Conference on Machine Learning-Volume 32*, pp. 1188–1196, Beijing, China, June 2014.
- [24] L. Page, S. Brin, R. Motwani et al., "The pagerank citation ranking: bringing order to the web," Technical report, Stanford InfoLab, Stanford, CA, USA, 1998.
- [25] B. Yang, W.-t. Yih, X. He et al., "Embedding entities and relations for learning and inference in knowledge bases," in *Proceedings of the International Conference on Learning Representations (ICLR)*, San Diego, CA, USA, May 2015.
- [26] W. Liu, J. Liu, M. Wu et al., "Representation learning over multiple knowledge graphs for knowledge graphs alignment," *Neurocomputing*, vol. 320, pp. 12–24, 2018.
- [27] M. A. Rodríguez and M. J. Egenhofer, "Determining semantic similarity among entity classes from different ontologies," *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 2, pp. 442–456, 2003.
- [28] Z. Sun, W. Hu, and C. Li, "Cross-lingual entity alignment via joint attribute-preserving embedding," in *Proceedings of the 16th International Semantic Web Conference*, pp. 628–644, Vienna, Austria, October 2017.
- [29] M. Schlichtkrull, T. N. Kipf, P. Bloem et al., "Modeling relational data with graph convolutional networks," in *Proceedings of the European Semantic Web Conference*, pp. 593–607, Portorož, Slovenia, June 2018.
- [30] T. Trouillon, J. Welbl, S. Riedel et al., "Complex embeddings for simple link prediction," in *Proceedings of the 33rd International Conference on International Conference on Machine Learning-Volume 48*, pp. 2071–2080, New York, NY, USA, June 2016.
- [31] X. Han, S. Cao, X. Lv et al., "Openke: an open toolkit for knowledge embedding," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 139–144, Brussels, Belgium, November 2018.