JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 39, 291-303 (2023)
DOI: 10.6688/JISE.202303_39(2).0003

Inconsistency Detection in Knowledge Graph with Entity and Path Semantics*

ZHI-YU HONG[1] AND ZONGMIN MA[1,2,+]
[1]*College of Computer Science and Technology*
Nanjing University of Aeronautics and Astronautics
Nanjing, 211106 P.R. China
[2]*Collaborative Innovation Center of Novel Software Technology and Industrialization*
Nanjing, 210023 P.R. China

Knowledge Graphs (KGs), which contain rich relational information, have been widely utilized in various tasks. However, there may exist inconsistent facts in KGs, especially in automatically constructed large-scale KGs. To address this problem, we innovatively propose an entity&paths semantics based multi-classification model to solve the problem of inconsistency detection. It synthesizes the internal semantic information both in entity and relation level of the KG to measure the association strength between triples so that different kinds of inconsistencies can be accurately detected. We conduct experiments in the real-word dataset FB15k (from Freebase) and the results show that our approach achieve significant and consistent improvement compared to existed advanced approaches, confirming the capability of our framework in knowledge graph inconsistency detection.

Keywords: knowledge graph, knowledge graph quality, inconsistency detection, entity& path semantics, multi-classification

1. INTRODUCTION

Knowledge Graphs (KG) are meant to contracture effective structured information and have become a crucial backbone of many emerging research and applications, such as knowledge retrieval, recommendation, and decision making [1-3]. A typical KG usually depicts individuals in the real world and their relationships as multi-relation data and expresses facts in the triple form of <*h*, *r*, *t*>, where *h* and *t* denote head entity and tail entity, and *r* denote a relationship between head and tail.

Recent years, with the widespread popularization of downstream applications of knowledge graphs, the construction of large-scale knowledge graphs has become an increasing urgent matter. Currently, knowledge graph construction mostly adopts automatic construction or tries to integrate existing knowledge graphs instead of early manual work [4] with the surging of data, such as Knowledge Vault [1], FreeBase [2] and DBpedia [3]. Nevertheless, due to the pattern or instance level rush or data conflicts during the fusion of different knowledge bases, the knowledge graphs we get inevitably exist inconsistent quality problem. For example, one of the state-of-the-art relation extraction approach achieves approximately 60% precision [19, 20]. In this case, it is crucial to take measures to ensure the quality of KGs.

Our objective is to detect possible noise and conflicts existing in large-scale KGs, while classifying the inconsistencies according to the manifestations of them. Most con-

Received June 24, 2021; accepted August 31, 2021.
Communicated by Shyi-Ming Chen.
+ Corresponding author: zongminma@nuaa.edu.cn.
* This work was supported by the National Natural Science Foundation of China (62176121 and 61772269) and the Basic Research Program of Jiangsu Province (BK20191274).

ventional methods rely on type information of entities, and attempt to spot violations of typical usage pattern of a relation or the underlying ontology constraints [8, 9, 18]. While these approaches ignore the cases of wrong instances of correct types, like the triple (*Grorge W. Bush*, *president*, *Hillary Cliton*), it could be recognized as correct fact with such approaches. Knowledge representing learning (KRL) methods meant to transform entities and relations into lower-dimensional, fixed-sized vectors, are also used to detect errors in KGs [5-7]. However, KRL encounter a bottleneck when facing to intricate paths between entities, particularly in large-scale KGs.

In general, detecting inconsistencies in large-scale KGs is still a challenging work with three main challenges that need to be addressed for effectively inconsistency detection in KGs; (1) Symbolic and logical reasoning cannot find out the inconsistencies in large-scale discrete KGs [18]; (2) It's hard to grasp the association strength among triples; (3) No golden standard to learn or observe the pattern of false knowledge.

We attempt to address the above challenges with a novel entity&path-semantics based framework regarding the discrimination of inconsistencies as a multi-classification task where each output corresponds with one kind of inconsistency to solve this problem. We investigate multifarious errors in existing knowledge graphs [4], and separate possible inconsistencies into three types, *i.e.*, entity-related inconsistency, relation-related inconsistency and type-related inconsistency. Our work is based on the following insight, both entities and relations in KG maintain a large quantity of semantic information [11-14], where highly correlated information can be leveraged to detect the inconsistent triples. To make the inconsistency detection more universal and accurate, we combine the entity pairs and paths between them to achieve better semantic information representation. In the entity level, we propose three kinds of entity strength representation considering semantic related information between entities, namely entity semantic representation, entity similarity and relative path confidence, to measure the strength of entity association in an all-round way. In the path level, we introduce the concept of path support based on various paths of different lengths, which indicates the degree of certainty whether the relation connecting the entity pair is the correct one or not.

In experiments, we evaluate our framework on datasets with different forms of inconsistencies. Specially, we compare the performance of four classifiers on inconsistency detection. The experimental results demonstrate that our models achieve the best performances in all kinds of datasets, which confirm the capability of semantic information of entity and path in inconsistency detection. The main contributions of this work are concluded as follows:

- We propose a novel entity&path semantics-based framework for knowledge graph inconsistency detection that makes full use of semantic information hidden in the entity pairs and paths between them.
- We evaluate our models on different datasets extends from benchmark and experimental results show promising performances on all datasets compared to the state-of-art methods.

2. RELATED WORK

Approaches to detect errors in knowledge graphs have developed rapidly since last years. Knowledge representation learning are one of the state-of-art methods for the detec-

ion work where entities and relations are transformed into lower-dimensional, continuous vector spaces to gain better semantic representation of triples. TransE introduced by Borders *et al*. in [5] is one of the first family members of Translation. Its main idea is that the embedding vector of the subject and predicate approximately equals the vector of object for a particular triple <*s*, *p*, *o*>, that is, $s+p \approx o$. Although TransE can perform well when dealing with simple relationships, it loses its advantages facing to multi-relation cases where the same relationship can occur between different entities and various relations can associate with the same entity pair. Xie *et al.* [6] improve the knowledge representation through injecting a triple confidence measure considering three kinds of triple confidence involving internal structure information in KG. However, it pays more attention to relationships, ignoring related information associated with entities and lose effect in only entity related inconsistent case.

Paulheim *et al.* [9] probe into the statistical distribution of the types and relations with SDValidate, finding the incorrect objects which are incompatible with the characteristic distribution of a given property. SDValidate pays more attention to type assertions, which does not work in the case of wrong entities with correct types or the type information is insufficient.

Graph model is another noteworthy field highly related to inconsistency detection where entity is abstracted into a node and the edge means a relationship from entity *h* to *t*. In [11], Jia *et al.* introduced trustworthiness measurement model judging a triple's trustworthiness by three aspects, correlation between subject and object of a triple, translation variance of relation vectors and acknowledgement from relevant triples. However, the absence of path-to-path calculation leads to poor performance in relation-related inconsistency. COPAAL [12] infers that predicate in triples carries much mutual information which can make contribution to the validation of facts. However, the RDFS scheme information which helps for path searching is not accessible in many cases, and this will make it out of action.

Most existing inconsistency detection methods look into potential structural information of entities or relations, however, they either do require the hold of rich schema information [9, 12], or are biased to a special category of inconsistency [6, 11]. In this paper, we aim to introduce an entity&path semantics based multi-classification model free from the effect of schema and recognize possible errors with the same force. We verify the effect of the semantic information of entity pairs and paths between them for a given triple based on graph model for better knowledge representation. We consider making comprehensive combination of entity and path semantic with the help of graph representation structure to verify the consistency of triples in knowledge graphs. For the sake of the discrimination of different kinds of inconsistencies, the inconsistency detection task is regarded as a special case of multi-classification measurement, in which different label denotes various kinds of inconsistencies or consistent triples. Under the guidance of semantics of entity and relation, our model can identify and distinguish all possible inconsistencies with a more flexible and universal way than above methods.

3. METHODOLOGY

We establish a multi-classification model that identify various kinds of inconsistencies based on the semantic information of entities, as well as the information hidden in the

paths between nodes representing entities. Differing from conventional methods which detect false triples using one specific type of feature, we not only strengthen the calculation of semantics of entities but also take comprehensive features of triples in KG, so that we can recognize all types of inconsistent triples.

3.1 Problem Formulation

In this section, we first give the notations used throughout this paper. Given a triplet (h, r, t) in KG expressed as G, we consider the head and tail entities $h, t \in E$ and the relation $r \in R$, where E and R represent the set of entities and relations in G. Specially, we denote the set of types of entities as T (with $T \subseteq E$) for identifying inconsistencies in more detail. According to different manifestations of possible existing conflicts, there are three types of inconsistent knowledge whose explanation listed as follows.

Definition 1 (Relation-related Inconsistency): An inconsistent knowledge related to relation means the relation connecting head and tail is false, *i.e.*,

$$RI = \{(h, r, t) \mid h \in E \wedge t \in E \wedge (h, r, t) \notin G\}.$$

Definition 2 (Entity-related Inconsistency): An inconsistent knowledge related to entities is a false triple with one wrong entity in G, *i.e.*,

$$EI = \{(h', r, t) \mid t \in E \wedge r \in R \wedge (h', r, t) \notin G\} \cup \{(h, r, t') \mid h \in E \wedge r \in R \wedge (h, r, t') \notin G\}.$$

Definition 3 (Type-related Inconsistency): An inconsistent knowledge related to type is an erroneous mapping from entity to type, its formula is as follows,

$$TI = \{(h', r, t) \mid t \in T \wedge r \in R \wedge (h', r, t) \notin G\} \cup \{(h, r, t') \mid h \in E \wedge r \in R \wedge (h, r, t') \notin G\}.$$

3.2 Entity & Path Semantics

To make the acquisition of semantic features of entities and paths more universal and practical, we only consider the internal structure after KG construction in our framework, and we propose two methods for semantic information of entities and paths respectively in the following section.

3.2.1 Entity strength

We assume that a triple will be considered to have better intern correlation strength if h behaves semantically close to t. Those with higher correlation strength will get higher trust. In the following subsections, we will introduce how to qualify the semantic information of entity for a given triple.

(1) Entity Semantic Representation

We assume that an entity t should be thought more important for h if more information is transmitted from h flowing to t for a given entity pair (h, r, t). Specially, we follow the

path-constraint resource allocation (PCRA) [15] and ResourceRank (RR) [11] to measure the semantics information of t projected from h.

Formally, given an entity pair (h, t), the resource associated with h will be emitted to t through several paths (p_1, p_2, …, p_l). Given that there are probably multiple entities connecting one relation, each path p_i is represented as $h \xrightarrow{r_1} E_1 \xrightarrow{r_2} E_2 \xrightarrow{r_3} \ldots \xrightarrow{r_{n-1}} E_{k-1} \xrightarrow{r_k} t$, where E_i denotes the entity set at the ith step, k means the number of relations included on the path p_i. For each entity e (including head entity and tail entity) on the path p_i, the semantic value $S(e)$ can be calculated as follows:

$$S(e) = (1-\theta) \sum_{e' \in E_{i-1}(\cdot,e)} \frac{S(e') \cdot W_{e'e}}{|E_i(e', \cdot)|} + \theta. \tag{1}$$

Here $E_{i\text{-}1}(\cdot, e)$ denotes the direct predecessor of e through r_i, $E_i(e', \cdot)$ denotes the direct successors of e^i via r_i and $W_{e'e}$ is the weight of edge $e' \rightarrow e$ depending on the number of relations connecting e' and e. In order to avoid the influence of the closed loops when resource flows, we use a hyper-parameter θ representing that the resource will flow to next entity $e \in E_i(e', \cdot)$ with the same probability. The semantic value $S(t)$ is considered as the semantic information implied in the entity pair (h, t).

(2) Entity Similarity Representation

For further enhancing the association strength of the entity pair (h, t), we propose entity similarity representation expressing the semantic distance of h and t according to their learned embeddings. Following the translation rule in [5], for the entity pair (h, t) and path $p_i = (r_1, r_2, \ldots, r_k)$ we define entity path similarity function as follows:

$$SP(h, t, p_i) = \cos(h - t, p_i) = \cos(h - t, (r_1 + r_2 + \ldots + r_k)). \tag{2}$$

Here cos(·) denotes cosine-similarity function. Since we assume that head entity embedding plus path embedding should be similar as tail entity embedding, the lower $SE(h, t, p_i)$ is, the more similar (h, t) gets. The entity similarity representation through l paths is expressed as follows:

$$SE(t) = \frac{1}{l} \sum_{i=1}^{l} SP(h, t, p_i). \tag{3}$$

(3) Relative Path Confidence

In this subsection we introduce a supplement measurement in case of the right entity pair with wrong relation happens. Relative path confidence (RP) is meant to capture the characteristics between entity and relation considering their integration degree. Triples with higher entity/relation integration degree are more likely to be consistent than those with lower ones. The relative path confidence of a triple is defined as follows:

$$RP(r \mid e) = \frac{|Co(e \wedge r)|}{|O(e)|}. \tag{4}$$

Here cos(·) means co-occurrence function and $O(\cdot)$ means the number of occurrences of
• $RP(r|e)$ stands for the probability of e and r share the same nature. As the auxiliary feature

for entity strength, the relative path confidence describes the frequency of the combination of head/tail and relation, helping to identify various types of inconsistencies.

3.2.2 Path support

To identify multiple inconsistencies, we utilize multi-steps paths between *h* and *t* for the given triple to express the semantic relevance and the complicated inference patterns of different triples. Each path can provide support for the existence of relation *r* to varying degrees [12]. Consequently, integrating all the supporting paths can strongly corroborate the correctness of *r* between *h* and *t*. However, since not all paths make sense, the two key factors for exploiting path support are reliable path searching and path support measurement. We show our solutions in the following subsections:

(1) Reliable Path Searching

We introduce path semantic confidence (PS), which measures the reliability of a relation path to the direct relation *r* based on the semantic relevance of the path with *r*, under the inspiration of [11] to pick up the most significant paths. The basic idea behind the PS is that the path embedding should be similar as the target triple embedding.

Formally, given a target triple (*h*, *r*, *t*) and the path $p = (h \xrightarrow{r_1} e_1 \xrightarrow{r_2} e_2 \xrightarrow{r_3} ... \xrightarrow{r_l} t)$ connecting *h* and *t*, the path semantic confidence is defined as follows:

$$PS(p) = \frac{1}{l}\left(\sum_{i=1}^{l} \frac{r_i \cdot r}{\| r_i \| \| r \|} + \frac{(t-h)\cdot\left(\sum_{i=1}^{l} e_i\right)}{\| t-h \| \| \sum_{i=1}^{l} e_i \|}\right). \tag{5}$$

In contrast to [11] in which takes the embedding distance of head entity and tail entity with entities on the path into account separately, the second part above have reduced the impact of entities on the relation path in the way of comparing the semantics of entities on it with the difference of *h* and *t*. In this way, both entities and relations on path have influenced our choice in varying degrees.

(2) Path Support Measurement

Once the paths are selected, it is necessary to get the support for the direct relation *r* of each path. We base our computation of the path support between relation path p_i and relation *r* on the normalized pointwise mutual information (NPMI) [16]. In a formal manner, given a target triple (*h*, *r*, *t*) and the relation path $p_i = (h \xrightarrow{r_1} E_1 \xrightarrow{r_2} E_2 \xrightarrow{r_3} \ldots \xrightarrow{r_l} t)$ in the multiple relation graph, we first give some common notations used in the computation as follows:

Notation 1: $E(T)$: a set of entities whose type is *T*, *e.g.*, $Donald\_Trump \in E(Person)$.
Notation 2: $\gamma(e_s, e_o \mid l)$: a relation path of length *l* connecting entity e_s and entity e_o through the specific path $p = (r_1, r_2, \ldots, r_l)$.

Firstly, we introduce the probability $P(r)$ of the relation *r* linking the instances of type T_h and T_t:

$$P(r) = \frac{\sum_{e_s \in E(T_h),\ e_o \in E(T_t)} |(e_s, r, e_o)|}{|E(T_h)| \cdot |E(T_t)|}. \tag{6}$$

Here T_h and T_t denote the type of *h*, *t* respectively and (e_s, r, e_o) is the triple in *G*.

In the second step, we would like to obtain the probability $P(p_i)$ of the relation path $p_i = (h \xrightarrow{r_1} E_1 \xrightarrow{r_2} E_2 \xrightarrow{r_3} \cdots \xrightarrow{r_l} t)$ with the equation as below, it shows the possible occurrence of the specific relation path between the pairs of instances of T_h and T_t.

$$P(p_i) = \frac{\sum_{e_s \in E(T_h),\ e_o \in E(T_t)} \gamma(e_s, e_o \mid l)}{|E(T_h)| \cdot |E(T_t)| \cdot \prod_{i=1}^{l} |E_i|} \tag{7}$$

In the next step, we define the joint probability $P(p_i, r)$ representing the co-occurrence of the relation path p_i and the direct relation r as follows:

$$P(p_i, r) = \frac{\sum_{e_s \in E(T_h),\ e_o \in E(T_t)} |\gamma(e_s, e_o \mid l) \cap (e_s, r, e_o)|}{|E(T_h)| \cdot |E(T_t)| \cdot \prod_{i=1}^{l} |E_i|}. \tag{9}$$

Finally, based on those path calculations we can now approximate the NPMI of the relation path p_i and the direct relation r as follows:

$$NPMI(p_i, r) = \frac{ln\ P(p_i, r) - ln\ P(p_i) - ln\ P(r)}{-ln\ P(p_i, r)}. \tag{10}$$

The $NPMI(p_i, r)$ depicts the semantic information hidden in the relation path, which behaves as path support of p_i to the direct relation r for proving whether the given triple is a consistent one or not.

3.3 Implementation

In this section, we are about to present the implementation of the formal semantic model presented above. The process of entity level and path level shows as follows:

As for entity semantics representation, our implementation begins by identifying a set of entities in varying lengths of paths that make resources flow from *h* to *t*. For each entity, its path found out first. Then we can iterate the resources though the path until the resource retention value of *t* calculated. Besides, the dissimilarity of *h* and *t* on the path will be computed simultaneously with pre-trained embeddings. Subsequently the degree of combination of entity and corresponding relationship is retrieved as entity supplement feature for the validation of relation-related inconsistent cases with the help of triples that read in. Finally, all entity pair correlation strength measurements are transformed into the vector denoting the entity characteristics for the classifier.

In the path support section, as too far logical chain of paths will lead to a decrease in the effect of path inference [6], only those paths less than or equal to 4 are selected for further computations. Once the paths are selected their semantic distance to *t* are obtained according to the embeddings and the top *K* of semantics will be screened out. Based on the normalized pointwise mutual information [16], the probability of *t*, relation path p_i and their joint probability are gotten for representing the paths support between p_i and *r*. In the end, we get those top *K* paths support to the relation and convert them to the path feature for the classifier.

4. EXPERIMENTS

This section verifies the accuracy and reliability of the proposed scheme through simulation and comparison of the performance with several well-known schemes.

4.1 Datasets

In this paper, we apply FB15K [5] in the evaluation of our multi-classification-based inconsistency detection model. FB15K is a typical benchmark KB extracted from Freebase [2]. We follow the entity classification work in [13] with entity type information which can be used for type-related inconsistency detection. To this end, we add type information for the deleted 47 entities in [13] and remove the default type "/common/topic" to eliminate the negative influence of meaningless information. In terms of synthesized labelled negative facts, we employ the same methods represented in [6] by replacing one element in the given relation triple (*h*, *r*, *t*) to generate a negative triple.

Following this protocol, we build a corpus for our experiments using FB15k and its type information where entity-related inconsistency, relation-related inconsistency and type-related inconsistency each account for 25%, 12.5%, 12.5% with one of three kinds of fake ones may be constructed for each true triple in a quantitatively balanced way. The statistics of the datasets are listed in 0.

Table 1. Statistics of datasets.

DATASET	#REL	#ENT	#TRAIN	#VALID	#TEST	#TYPE-INFO
FB15K	1346	14,951	483,142	50,000	59,071	168,584

Datasets	Ent-related Inconsistency (FB15k-EI)	Rel-related Inconsistency (FB15k-RI)	Type-related Inconsistency (FB15k-TI)	All kinds of Inconsistency (FB15k-AI)
H#Neg triple	190,199	95,099	95,099	380397

4.2 Experimental Settings

In experiments, we evaluate our entity&path-semantics inconsistency detection models with four different classification strategies, *i.e.*, SoftMax as classifier, Random Forest (RF), Deep Neutral Network (DNN) and Gradient Boosting Decision Tree (GBDT). We perform inconsistency detection tasks on the two different types features of entity-semantics and path-semantics separately, on this basis, the fusion of the two characteristics is considered as a global feature to compare the improvement effect of different forms of semantics for inconsistency detection, and the strongest classifier is selected.

We implement TransE [5] and CKRL [6] as baselines since CKRL has achieved great progress considering local and global path information and TransE has always been a classic baseline for error detection task.

4.3 Evaluation Protocol

For evaluation follow [17], we use mean average precision, mean average recall, macro-F and micro-F which are commonly used in multiclass classification as the evaluation method for inconsistency detection from different aspects. Besides, we utilize confusion matrix to more intuitively observe the performance of the model in each inconsistency category and Receiver Operating Characteristic Curve (ROC) to measure the generalization ability of the model.

4.4 Experimental Results

(1) Comparison with baselines

We look forward to finding out the best performing classifier distinguishing inconsistencies as detailed as possible. For that purpose, we compare four classification strategies based on entity and path semantics with baselines on datasets doped with various kinds of inconsistencies. In comparison with the same task of utilizing TransE and CKRL methods, we demonstrate the results of inconsistency detection shown in 0.

Table 2. Evaluation results on the Knowledge Graph inconsistency detection

Datasets	**FB15k - EI**		**FB15k - RI**		**FB15k - TI**		**FB15k - AI**	
Metric	Accuracy	F1score	Accuracy	F1score	Accuracy	F1score	Accuracy	F1score
TransE	0.813	0.860	0.796	0.834	0.786	0.843	0.759	0.827
CKRL	0.837	0.854	0.840	0.846	0.813	0.823	0.799	0.846
SoftMax	0.974	0.973	0.939	9.938	0.956	0.960	0.956	0.954
RF	0.978	0.957	0.941	0.935	0.960	0.941	0.973	0.981
GBDT	**0.982**	**0.969**	**0.950**	**0.947**	**0.963**	**0.952**	**0.974**	**0.975**
DNN	0.852	0.851	0.812	0.783	0.891	0.867	0.812	0.832

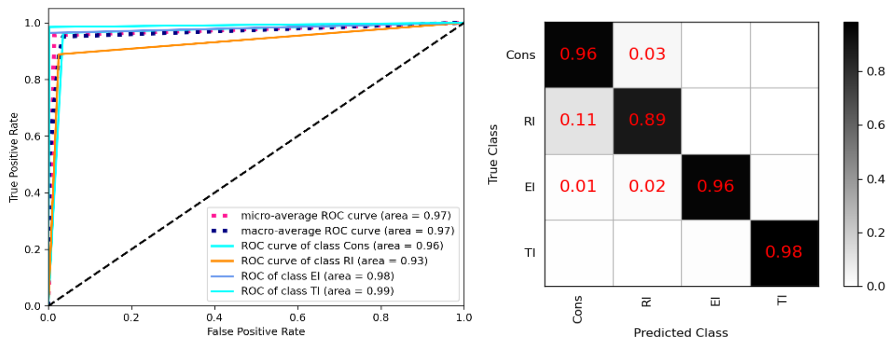
(i) Our best multi-classification model has achieved impressively *95%~98%* in accuracy, showing approximately *13%~15%* improvements than baselines on all datasets with varying kings of inconsistencies. It directly confirms the capability of semantic information of entity and path in checking inconsistent knowledge regarding entities and relations in KGs; (ii) Comparing evaluation results with different datasets, we find that most approaches have obtained a better overall performance in *FB15k-EI*, but behaves worse with *3%~5%* down in *FB15k-RI* and *FB15k-AI*. It indicates that Entity and path semantics can capture the difference of entity pairs more obviously; (iii) Comparing the four strategies taking entity&path semantic information as features, GBDT have outperformed others with an improvement of *4%~14%* in accuracy, which means GBDT is more adaptable to semantic information. In the next subsection, we will use GBDT to further verify the discrimination strength of semantic information of entity and path for detecting and distinguishing various conflicts.

(2) Discrimination strength of entity and path semantics

To verify whether the semantic information of entity and path is valid for discriminating various types of inconsistent knowledge, we choose SoftMax to do the discrimination strength and generalization analysis on the test set of *FB15k-AI* since it obtained the average performance so that we can see more intuitively the discrimination of entity and path semantics. The detection results for each inconsistency shows in Fig. 1.

In Fig. 1 (a), we can observe that our model based on semantic information can well recognize positive knowledge from datasets with a large amount of noise. In contrast, a considerable portion of relation-related inconsistency are misjudged as *Cons*. From Fig. 1 (b), it is observed that our models reach a higher false positive rate in the case of detecting *RI*, which correspond with Fig. 1 (a). It may be due to the uncertainty and incompleteness

in path searching caused by noises and limited path length. Taking longer and better paths into consideration will partially deal with this problem while at the price of extraordinary time consuming. To verify this end, we analyze the impact of different forms of semantic information in the following subsection.



(a) Confusion matrix (b) ROC

Fig. 1. Evaluation results on differentiating various types of inconsistencies.

(3) Effects of single semantics

We separate each form of semantic information of entity and path as an independent model to do the inconsistency detection, in which we use precision and recall as evaluation measurement to explore the ability of each model to recall the positive samples from candidate triples doped with a great deal of noises. Follow results of comparative test, we choose GBDT classifier since it gets the best performance on all datasets. 0 demonstrates the results of single semantics analysis.

We can observe that (1) the precision obtained by each model exceeds *80%*. It confirms the effectiveness of semantic information of entity and path, each of whom can be utilized for detecting inconsistent triples of multiple categories in KGs; (2) Integrating both entity semantics and path semantics helps GBDT achieve an extraordinary precision of over *95%*, which indicates that entity and path could play a complementary role to each other; (3) Comparing the performance of these two models in different datasets, it can be found that GBDT (ent) outperforms GBDT (path) on *FB15k-EI* while GBDT (path) obtain a higher level in *FB15k-RI*, which also verifies the effect of entity and path semantics in entity-related inconsistency resp. relation-related inconsistency. Moreover, the result of GBDT (path) on *FB15k-RI* shows a little lower than GBDT (ent) on *FB15k-EI*. It may be because of the diversified strategies we take in entity semantic representation. Therefore, we consider increasing the means of expression of path information in the future work.

Table 3. Evaluation results of effects of single semantics.

Datasets	**FB15k - EI**		**FB15k - RI**		**FB15k - TI**		**FB15k - AI**	
Metric	Precision	Recall	Precision	Recall	Precision	Recall	Precision	Recall
GBDT (ent)	**0.934**	0.921	0.813	0.801	**0.915**	0.894	**0.921**	0.918
GBDT (path)	0.810	0.798	**0.904**	0.897	0.847	0.823	0.889	0.853
GBDT (ent+path)	**0.991**	**0.973**	**0.957**	**0.938**	**0.972**	**0.953**	**0.977**	**0.969**

5. CONCLUSION AND FUTURE WORK

In this paper, we aim to eliminate the inconsistent knowledge which could harm the knowledge-driven learning tasks and applications. To this end, we explore semantic information expression methods of entity and paths between entity pairs which can be used for inconsistency detection in KG. Furthermore, we view multiple inconsistencies as multiclassification problem corresponding to entity-related inconsistency, relation-related inconsistency and type-related inconsistency, and establish an entity&path semantics based multi-classification model to identify and distinguish concrete types of conflicts, where each inconsistency corresponds with different kinds of correction methods. In our experiments, we compare four classification strategies to acquire the best classifier in detect different types of inconsistencies and evaluate the effectiveness of semantic information of entity and path. Experimental results indicate that our entity&path semantics based multi-classification framework achieve better performance than other baselines on the same task of inconsistency detection, which confirms the capability of our approaches of capture the semantic information hidden in entity and paths.

In the experiment of entity and path semantic discrimination, it can be found that the error rate of relation-related inconsistency is higher than other inconsistencies. This may be due to the deviation of path selection or path information calculation. In the future work, we will take other path semantic information representation method into consideration. Furthermore, we have recognized various types of inconsistencies, which lays the foundation for the correction of whom. We will explore to correct inconsistencies in KG according to their inconsistent categories.

REFERENCES

1. X. Dong, E. Gabrilovich, G. Heitz, *et al.*, "Knowledge vault: A web-scale approach to probabilistic knowledge fusion," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 601-610.
2. K. Bollacker, C. Evans, P. Paritosh, *et al.*, "Freebase: A collaboratively created graph database for structuring human knowledge," in *Proceedings of ACM SIGMOD International Conference on Management of Data*, 2008, pp. 1247-1250.
3. J. Lehmann, R. Isele, M. Jakob, *et al.*, "Dbpedia – a large-scale, multilingual knowledge base extracted from wikipedia," *Semantic Web*, Vol. 6, 2015, pp. 167-195.
4. H. Paulheim, "Knowledge graph refinement: A survey of approaches and evaluation methods," *Semantic Web*, Vol. 8, 2017, pp. 489-508.
5. A. Bordes, N. Usunier, A. Garcia-Duran, *et al.*, "Translating embeddings for modeling multi-relational data," *Neural Information Processing Systems*, 2013, pp. 2787-2795.
6. R. Xie, Z. Liu, F. Lin, *et al.*, "Does William Shakespeare really write Hamlet? Knowledge representation learning with confidence," in *Proceedings of AAAI Conference on Artificial Intelligence*, 2018, pp. 4954-4961.
7. Q. Wang, Z. Mao, B. Wang, *et al.*, "Knowledge graph embedding: A survey of approaches and applications," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 29, 2017, pp. 2724-2743.

8. J. Pujara, H. Miao, L. Getoor, *et al.*, "Knowledge graph identification," in *Proceedings of International Semantic Web Conference*, 2013, pp. 542-557.
9. H. Paulheim and C. Bizer, "Improving the quality of linked data using statistical distributions," *International Journal on Semantic Web and Information Systems*, Vol. 10, 2014, pp. 63-86.
10. A. Melo and H. Paulheim, "Detection of relation assertion errors in knowledge graphs," in *Proceedings of the 9th International Conference on Knowledge Capture*, Vol. 22, 2017, pp. 22:1-22:8.
11. S. Jia, Y. Xiang, X. Chen, *et al.*, "Triple trustworthiness measurement for knowledge graph," in *Proceedings of the World Wide Web Conference*, 2019, pp. 2865-2871.
12. Z. H. Syed, M. Röder, and A. C. N. Ngomo, "Unsupervised discovery of corroborative paths for fact validation," in *Proceedings of International Semantic Web Conference*, 2019, pp. 630-646.
13. R. Xie, Z. Liu, J. Jia, *et al.*, "Representation learning of knowledge graphs with entity descriptions," in *Proceedings of AAAI Conference on Artificial Intelligence*, 2016, pp. 2659-2665.
14. P. Lin, Q. Song, J. Shen, *et al.*, "Discovering graph patterns for fact checking in knowledge graphs," in *Proceedings of International Conference on Database Systems for Advanced Applications*, 2018, pp. 783-801.
15. Y. Lin, Z. Liu, H. Luan, *et al.*, "Modeling relation paths for representation learning of knowledge bases," *arXiv Preprint*, 2015, arXiv:1506.00379.
16. G. Bouma, "Normalized (pointwise) mutual information in collocation extraction," in *Proceedings of International Conference of the German Society for Computational Linguistics and Language Technology*, 2009, pp. 31-40.
17. M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing & Management*, Vol. 45, 2009, pp. 427-437.
18. Z. Ma, F. Zhang, L. Yan, and J. Cheng, *Fuzzy Knowledge Management for the Semantic Web*, Springer, Berlin, Heidelberg, 2014, pp. 157-180.
19. Y. Lin, S. Shen, Z. Liu, *et al.*, "Neural relation extraction with selective attention over instances," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 2016, pp. 2124-2133.
20. S. Heindorf, M. Potthast, B. Stein, *et al.*, "Vandalism detection in wikidata," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 2016, pp. 327-336.

Zhi-Yu Hong is currently pursuing his master degree in the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China. His research interests include RDF data management and knowledge graph.

Zongmin Ma is a Full Professor at Nanjing University of Aeronautics and Astronautics, China. His research interests include big data, knowledge engineering, and computational intelligence. He has published more than two hundred papers and five monographs with Springer on these topics. He is a Fellow of the IFSA.