IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received November 22, 2020, accepted December 18, 2020, date of publication December 24, 2020, date of current version January 7, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3047086

SUKE: Embedding Model for Prediction in Uncertain Knowledge Graph

JINGBIN WANG[1,2], KUAN NIE[1,2], XINYUAN CHEN[3], AND JING LEI[1,2]
[1]College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350116, China
[2]College of Software, Fuzhou University, Fuzhou 350116, China
[3]Department of Information Engineering, Fuzhou Melbourne Polytechnic, Fuzhou 350121, China

Corresponding author: Kuan Nie (sxurjxynk@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61672159, and in part by the Industry-University-Research Collaboration in Fujian Province of China under Grant 2017H6008 and Grant 2018H6010.

ABSTRACT Graph embedding models are widely used in knowledge graph completion (KGC) task. However, most models are based on the assumption that knowledge is completely certain, and this is inconsistent with real-world situations. Although there are multiple studies on uncertain knowledge embedding tasks, they often use knowledge confidence to learn embedding and cannot make full use the structural and uncertain information of knowledge. This paper presents a new embedding model named Structural and Uncertain Knowledge Embedding (SUKE), which comprises two components: an evaluator and a confidence generator. For unknown triples, the evaluator learns the structural and uncertain information to evaluate its rationality and obtain a candidate set. The confidence generator then determines the confidence of the candidate set to achieve KGC. To verify the effectiveness of the proposed model, confidence prediction, triple evaluation, and fact classification tasks are performed on three data sets. Experimental results show that SUKE performs better than mainstream embedding methods. The model proposed in this paper can help advance the research on the embedding of uncertain knowledge graphs.

INDEX TERMS Artificial intelligence, knowledge representation, uncertain knowledge graph.

I. INTRODUCTION

Knowledge graph (KG) is a kind of directed graph, with nodes representing entities and edges representing the relations between entities, such as FreeBase [1], WordNet [2]. KG contain a large number of facts and play an important role in question-answering systems [3] and information extraction [4], among others. However, the process of utilizing KG is often accompanied by unavoidable problems, such as lack of knowledge.

As the scope of KG continues to expand, the above problem becomes increasingly serious. Prediction of new knowledge from existing knowledge has become an important task in the field of KGC. Existing embedding methods have given efficient performance on inference tasks, attracting much attention. The main idea is to encode entities and relations into vectors with machine learning methods and complete inference operations in a vector space.

The associate editor coordinating the review of this manuscript and approving it for publication was Alba Amato.

In recent years, more efforts have been devoted to obtaining high-quality vectors, such as using textual or image information of entities to enrich the vector representation. However, most embedding methods are based on the assumption that the knowledge is completely certain. In fact, uncertainty is the essence of knowledge, which indicates the probability of occurrence of facts.

In uncertain Knowledge graph (UKG) each fact has a confidence score. ConceptNet [5] calculates the confidence of each triple according to the number and reliability of knowledge sources. Never-Ending Language Learner (NELL) [6] uses the maximum expectation algorithm and semi-supervised learning to learn the confidence of each triple. UKG could be applied to a wide range of scenarios. For example, Probability KG Probase [7] provides the prior probability distribution of each concept behind the terms, which effectively supports short text understanding tasks involving disambiguation. In addition, [8] conducted related research on the question answering task of uncertain knowledge bases. Therefore, we should consider the uncertainty of knowledge when learning embedding.

Presently, there is limited research on the embedding of UKG. Miao *et al.* first proposed the reasoning model, Imperfect and Incomplete Knowledge Embedding (IIKE) [9]. Although IIKE has achieved good performance, it only considers each triple in isolation when calculating the confidence score. Furthermore, IIKE does not use the interrelated characteristics of the KG. The more prominent models are UnceRtain Graph Embedding (URGE) [10], proposed by Hu *et al.* in 2018, and Uncertain KG Embedding (UKGE) [11], proposed by Chen *et al.* in 2019. URGE is designed for uncertain networks and considers node proximity to generate node embedding. Although URGE can be extended to KGC, it cannot handle the task of embedding UKG well owing to the difference between uncertain networks and UKG. UKGE gives better performance than URGE. However, UKGE only uses knowledge confidence to learn embedding. This means that the structural information of knowledge, to a certain extent, is not fully utilized. UKGE also introduces probabilistic soft logic to generate unknown facts and related confidence scores, incorporating them into the training process, which is highly unreasonable. Because the rules extracted from UKG are inherently uncertain, this makes the inferred confidence score inaccurate. The direct incorporation of such facts into training will undoubtedly introduce unnecessary noise and reduce prediction accuracy. Therefore, effective integration of structural and uncertain information into embedding vectors could be meaningful.

To make full use of such information, this study decomposes the prediction task for UKG into two subtasks, namely, the rationality evaluation task and confidence prediction task. The former evaluates fact rationality based on the structural and uncertain characteristics of facts and screens out invalid facts to obtain candidate facts. The latter generates confidence for the candidates. In summary, this paper proposes SUKE, which includes an evaluator and a confidence generator. The evaluator defines structural and uncertain scores for each triple for the rationality evaluation task. In addition, the evaluator introduces probabilistic soft logic to learn unknown facts. The confidence generator generates triple confidence used for the confidence prediction task.

The contributions of this study are as follows:

- The method of fusing structural and uncertain information and the strategy of how unknown facts participate in training are proposed in the evaluator component, which effectively learns the structural and uncertain information of knowledge and enhances the ability of the model to identify positive and negative samples.
- The confidence generator component of the SUKE model uses the confidence of the positive example triples for training, thereby improving the fitting ability of the model and obtaining a more accurate confidence prediction.
- The SUKE model was evaluated using three datasets: a subset of ConceptNet [5] containing 15,000 entities (CN15k), a subset of NELL containing 27,000 entities (NL27k) and a subset of the Protein-Protein Interaction knowledge graph [12] containing 5000 entities (PPI5k). The experimental results show that SUKE performs better than other benchmark models. In particular, the lowest Mean Absolute Error (MAE) and Mean Square Error (MSE) were obtained in the confidence prediction task, and the highest Normalized Discounted Cumulative Gain (NDCG) was obtained in the triple evaluation task.

The remainder of this paper is structured as follows. First, relevant work is reviewed in Section II, definitions used in our model are provided in Section III, the model details are introduced in Section IV, the experiments are reported in Section V, and finally our conclusions and future research directions are reported in Section VI.

II. RELATED WORK

A. DETERMINISTIC KNOWLEDGE GRAPH

A KG is defined as $KG = (E, R)$ where E represents a set of entities and R represents a set of relations. KG could also be simply regarded as a set of triples (h, r, t), where $\{h, t\} \in E$, $r \in R$.

RESCAL [13] is a representative of tensor models. RESCAL uses tensor decomposition to capture complex relations, but it needs to calculate a large number of parameters, and what's more, matrix operations are of high cost. DistMult [14] simplifies the calculation process and achieves better performance.

Translation-based models are widely proposed, such as TransE [15]. TransE regards each relation as a conversion process between the head and tail entity embeddings. However, TransE cannot cope with complex relations. TransH [16] introduces a hyperplane for each relation and projects head and tail entities of triples onto relation-specific hyperplanes so that the same entity has different vector representations for different relations. Furthermore, such models include TransD [17], TransR [18], and RotatE [19].

Dettmers *et al.* propose the first convolutional neural network (CNN)-based model ConvE [20]. However, this model excludes local features among triple representations of the same dimensions. To address this problem, ConvKB [21] extracts local features of triples within the same dimensions with CNN's filters and has achieved very good results. Further, CapsE [22] improved ConvKB to achieve better performance.

B. UNCERTAIN KNOWLEDGE GRAPH

UKG provide a confidence score for each triple. The scores reflects the reliability of triples. In recent years, the development of relation extraction and crowdsourcing has promoted the construction of large-scale UKGs, such as ConceptNet, Probase [7] and NELL [6]. Fan *et al.* [9] first proposed the uncertain reasoning model, IIKE. IIKE uses a confidence probability formula for each triple, and the confidence of each triple was defined as the joint probability. It also modeled the prediction task as a conditional probability. In 2017, Hu *et al.* proposed URGE [10] which proposes a method based on

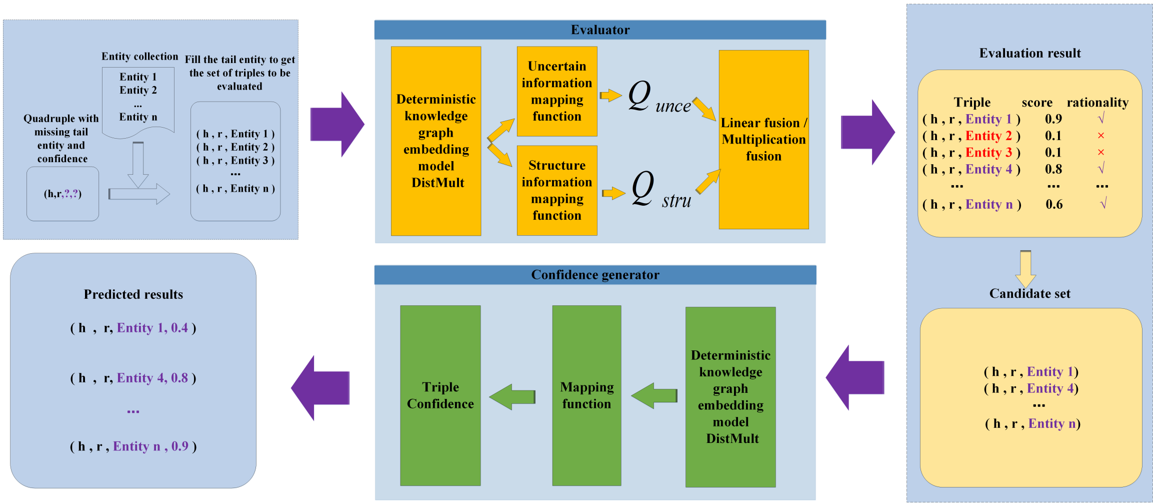


FIGURE 1. Prediction process. The input is a quadruple (h, r, ?, ?) with the tail entity and the confidence missing. SUKE first uses all the entities in KG to generate a set of triples, then evaluates the rationality of each triple with the evaluator, triples greater than a given threshold are added to the candidate set, and finally calculates the confidence of the candidate set with the confidence generator. For details of the evaluator and confidence generator, see Sections IV-A and IV-B.

matrix factorization to embed uncertain networks. However, this model only considers the proximity of nodes in the sparse network and only learns node embedding. Therefore, the application of URGE to UKG has certain limitations.

Another recent model proposed by Chen *et al.* in 2019 is UKGE [11]. UKGE defines the plausibility score for each triple, and proposes two variants.In addition, UKGE defines the calculation method of basic logic operations (logical conjunction $\wedge$, disjunction $\vee$, and negation $\neg$) based on the first-order logic rules, and correctly estimates the unknown triples and corresponding confidence in the uncertain knowledge base. Although UKGE has achieved good performance in the experiment, it only learns the conversion function from plausibility score for confidence, and to a certain extent does not make full use of the structure information. In addition, since the rules are inherently uncertain in the UKG, therefore, we should reasonably use the inferred unknown facts.

III. RELATED DEFINITIONS

A. UNCERTAIN KNOWLEDGE GRAPH(UKG) DEFINITION

UKG can be regarded as a weighted directed graph, defined as $UKG = (\epsilon, R, W)$ where ϵ represents a collection of entities, R represents a collection of relations, and W represents the confidence. UKG contains a large number of quadruples h, r, t, w where $\{h, t\} \in \epsilon$, $r \in R$, $w \in W$. A combination of (h, r, t) as the triple, and w is the confidence of the triple.

It should be noted that triples with low confidence in the noise knowledge graph will be regarded as noise, such as [23], [24], but in the UKG, we interpret the confidence as probability. A triple with low probability (low confidence) does not mean that the triple is incorrect.

B. EMBEDDING TASK OF UKG

The embedding task of UKG aims to learn low-dimensional vectors containing rich semantic information and confidence information for each entity and relation so that they can be used for succeeding tasks, such as fact classification, link prediction, and confidence prediction.

C. UKG PREDICTION TASK

For a prediction task with an uncertain knowledge graph, two questions need to be answered.The first question is: Is the triple (h, r, t) correct? The second question is: For a correct triple (h, r, t), what is the probability of this relation between entities? For example, (notice, related to, newspapers, 0.129) is an example of CN15k data, which means that the quantified correlation between notice and newspapers is 0.129. We need to evaluate the reasonableness of (notice, related to, newspapers) and obtain the confidence 0.129. Therefore, the prediction task of UKG can be divided into two categories: triple evaluation task and confidence prediction task. Triple evaluation could be defined as predicting a missing entity. Confidence prediction refers to predicting missing confidence given (h, r, t, ?).

IV. SUKE MODEL

Section III-C shows that the prediction task of UKG needs to evaluate the rationality of triples and give the confidence of reasonable triples. Based on DistMult energy score, this study designs two components: an evaluator and a confidence generator. The evaluator is used to evaluate the rationality of triples. Unreasonable triples will be removed by the evaluator model. The confidence generator is used to generate confidence for the candidate set. Fig. 1 shows SUKE's prediction process.

Considering that KG implies unknown facts, whose participation is meaningful in training, this paper introduces the probabilistic soft logic to obtain unknown facts, but the rules are inherently uncertain, leading to inaccurate confidence scores of the inferred facts.

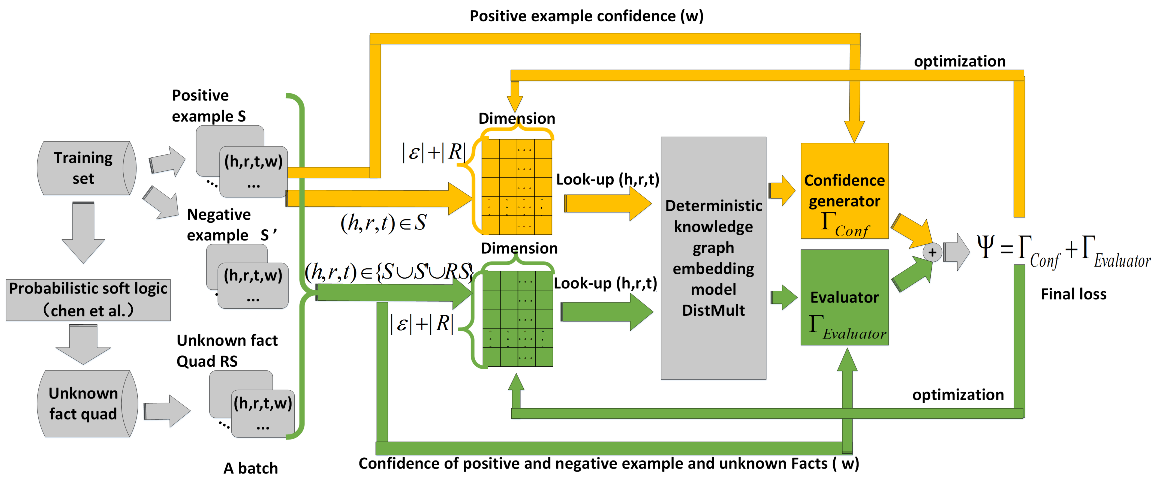


FIGURE 2. **Model training process. The evaluator learns the structural and uncertain information of the quadruple (please see Section IV-A for details). The confidence generator is designed to generate confidence for triples, which requires input of positive quadruples for training. Please see Section IV-A for details.**

Therefore, the unknown facts only participate in the training process of the evaluator. In addition, to avoid the mutual interference of embedding vectors between the evaluator and confidence generator, SUKE learns two vector representations for each entity and relation, which are used by the evaluator and confidence generator.The SUKE training framework is shown in Fig. 2.

A. EVALUATOR

1) EVALUATOR SCORE DEFINITION

In UKG, (h, r, t) carries not only structural information but also uncertain information. Although the uncertainty of a triple is not a criterion for judging whether a triple is reasonable, the uncertain information will help the model to better complete the evaluation task. Therefore, it is necessary to consider two kinds of information to evaluate the rationality of triples. Therefore, the evaluator score of each triple consists of two parts: structural score and uncertain score, denoted as Q_{stru} and Q_{unce}, respectively. A positive triple should have a high Q_{stru} and Q_{unce}.

In the task of embedding uncertain graphs, traditional deterministic models can still learn positive triples well. For example, deterministic models learn the uncertain positive triples (fox, isa, algonquian) and (wound, relatedto, bruise), and it predicts that (fox, relatedto, bruise) is wrong. However, during the training process, the energy scores of positive triple becomes smaller,while those of negative triples become larger,which is exactly contrary to the design intent of the evaluator score, at the same time, the uncertainty of knowledge needs to be modeled. so it is obviously unreasonable to directly use the energy score as the evaluator score.

To deal with this problem, this study learns mapping functions from energy scores E(h,r,t) to Q_{stru} and Q_{unce} separately. We first calculate energy scores of the triples with DistMult, and then obtained the triples Q_{stru} and Q_{unce} through different mapping functions. When optimizing, we will encourage the Q_{stru} of positive triples to be close to 1 and the Q_{stru} of negative triples to be close to 0; the Q_{unce} of the positive triples is encouraged to approach the true confidence w, and Q_{unce} of negative triples is encouraged to approach 0. Using this training method, the evaluator can distinguish between positive and negative examples in the uncertain knowledge graph.

The calculation method of Q_{stru} is shown in (1).

$$Q_{stru}(h,r,t) = \frac{1}{1+e^{-\phi_{stru}(E(h,r,t))}}. \tag{1}$$

where E(h,r,t) is the energy score of a triple obtained by the deterministic model, and ϕ_{stru} is the mapping function of the energy score of the triple to the structural score. The mapping function is shown in (2).

$$\phi_{stru}(E(h,r,t)) = P_{stru}\cdot E(h,r,t) + b_{stru}. \tag{2}$$

where P_{stru} and b_{stru} are the parameters.

The calculation method of Q_{unce} is shown in (3).

$$Q_{unce}(h,r,t) = \frac{1}{1+e^{-\phi_{unce}(E(h,r,t))}}. \tag{3}$$

where E(h, r, t) is the energy score of the triple obtained by the translation model, and ϕ_{unce} is the mapping function of the energy score of the triple to the uncertain score. The mapping function is shown in (4).

$$\phi_{unce}(E(h,r,t)) = P_{unce}\cdot E(h,r,t) + b_{unce}. \tag{4}$$

where P_{unce} and b_{unce} are the parameters.

In this study, the DistMult model is selected to calculate the energy score because of the following reasons. (1) DistMult employs a matrix operation, making the calculation process simple and fast. (2) Compared with other models, DistMult is simpler. (3) Compared with other models, DistMult achieves better performance in the deterministic KG. The energy score of the DistMult model is calculated using (5).

$$E(h, r, t) = h^{\mathrm{T}} diag(r)t. \tag{5}$$

where $diag(r)$ represents the diagonal matrix of relations, implying the relational interaction between entities.

2) FUSION METHOD OF STRUCTURAL SCORE AND UNCERTAIN SCORE

The structure score and uncertain score of the triples will be used to assess the rationality. This study focuses on two combination methods, namely, linear weighted fusion method and multiplication fusion method.

The linear weighted fusion method is shown in (6).

$$score = \alpha \cdot Q_{stru} + \beta \cdot Q_{unce}. \tag{6}$$

where $\alpha + \beta = 1$, the value of α and β determine the importance of the structural score and uncertain score, respectively. At the time $\alpha = \beta = 0.5$, the two scores are of the same weight. We use this setting in the experiments.

The multiplication fusion method is shown in (7).

$$score = \rho \cdot (1 - \rho) Q_{stru} \cdot Q_{unce}. \tag{7}$$

where ρ is the smooth hyperparameter. In this method Q_{unce} adjusts Q_{stru}. When Q_{unce} is small,it will lower the final score; otherwise the score will be boosted. When both the Q_{unce} and Q_{stru} are high, the calculated score will be high too; otherwise, it will be very small.

3) PROBABILISTIC SOFT LOGIC ENHANCEMENT METHOD

To enhance the learning ability of the evaluator, we use unknown facts to precipitate the training process. We introduce heuristic rules following the practice of Chen *et al.* [11] and use the probabilistic soft logic to obtain unknown facts. However, we find that the rules mined from UKG are inherently uncertain, inferring inaccurate unknown facts unavoidably. If the confidence is used directly, then unnecessary noise will be inevitably introduced. Therefore, during training, this study takes the confidence of unknown facts as the score of the evaluator and introduces dynamic parameters to improve the generalization ability of the model. The loss function of unknown facts is defined as (8).

$$\Gamma_{Fac} = \sum_{(h,r,t,w)\in RS} \left|\frac{Q_{stru}(h, r, t) + Q_{unce}(h, r, t)}{2} - w + \lambda\right|. \tag{8}$$

RS is the set of unknown facts, Q_{stru} and Q_{unce} are the structural score and uncertain score of the unknown facts, respectively. λ is the dynamic adjustment parameter, which can be obtained through learning.

4) EVALUATOR LOSS

The loss function of the estimator consists of positive loss, negative loss, and unknown fact loss, defined in (9)-(11), respectively.

$$\Gamma_{Evaluator} = \Gamma_{Pos} + \Gamma_{Neg} + \Gamma_{Fac}. \tag{9}$$

$$\Gamma_{Pos} = \sum_{(h,r,t,w)\in S} |Q_{stru} - 1|^2 + |Q_{unce} - w|^2. \tag{10}$$

$$\Gamma_{Neg} = \sum_{(h,r,t,w)\in S'} |Q_{stru}|^2 + |Q_{unce}|^2. \tag{11}$$

Γ_{Pos} represents a positive loss, Γ_{Neg} represents a negative loss, and Γ_{Fac} is the loss of unknown facts. S is the set of positive examples, and S' is the set of negative examples. Negative examples are generated by randomly replacing head or tail entities, and the confidence of the negative triple is set to 0.

Negative examples are as follows:

$$S' = \{(h_1, r, t)|h_1 \in \epsilon \setminus h\} \cup \{(h, r, t_1)|t_1 \in \epsilon \setminus t\}$$

B. CONFIDENCE GENERATOR

The confidence generator aims to generate confidence for triples. To reduce the complexity and parameters of the model, the confidence generator uses Q_{unce} to approximate the true confidence value w of triples. The confidence generator can be regarded as a confidence prediction model also based on the triple energy score and is different from the evaluator. The Q_{unce} of the confidence generator does not share parameters with the evaluator. In addition, the confidence generator does not require negative triples and unknown fact triples to participate in training.

The loss function of the confidence generator is defined as (12).

$$\Gamma_{Conf} = \sum_{(h,r,t,w)\in S} |Q_{unce} - w|^2. \tag{12}$$

C. MODEL LOSS FUNCTION

The overall model includes two components: an evaluator and a confidence generator. We simultaneously train two components, so the loss function of the model is defined as (13).

$$\Psi = \Gamma_{Evaluator} + \Gamma_{Conf}. \tag{13}$$

where $\Gamma_{Evaluator}$ is the loss of the evaluator(see Section IV-A for details) and Γ_{Conf} is the loss of the confidence generator model (see Section IV-B for details).

V. EXPERIMENT

A. DATA SET

To evaluate our model, we use three public datasets: CN15k, NL27k, and PPI5k. CN15k is a subset of ConceptNet, containing 15,000 entities and 36 relations. NL27k is a subset of NELL with 27,221 entities and 404 relations, which is larger and more complex compared to CN15k. PPI5k is a subset of STRING, which describes the reaction possibility among proteins. Compared with CN15k and NL27k, PPI5k has the fewest entities and relations, and correspondingly its quadruples are the densest. Furthermore, Chen *et al.* [11] heuristically created rules for the three data sets and used the logical rules to mine potential fact triples.

The three data sets contain a large number of quaternions (head, relationship, tail, confidence), and the confidence value range is (0,1). For example, (twitte, competitionswith, facebook, 0.859) is a fact in the NL27k dataset.

Dataset statistics are shown in Table 1.

TABLE 1. Statistics of the dataset.

DataSet	$\|\epsilon\|$	$\|R\|$	|Re.fact|	Avg(s)	Std(s)	|rule|	|Un.fact|
CN15k	15,000	36	241,158	0.629	0.232	3	7927
NL27k	27,221	404	175,412	0.797	0.242	4	32513
PPI5k	5000	7	271,666	0.415	0.212	1	85299

$|\epsilon|$ is the number of entities, $|R|$ is the number of relationships, |Re.fact| is the number of facts contained in the data set, Avg(s) is the average of confidence of all facts in the data set, Std(s) is the standard deviation score of all facts in the data set, |rule| is the number of rules used, and |Un.fact| is the number of Unknown facts deduced by rules. The rules are consistent with [11]. The storage link of the three data sets is: https://github.com/NKhandsome/SUKE/.

B. EXPERIMENTAL SETUP

This study uses the grid search method to determine the optimal parameters in the following set: learning rate lr $\in$ {0.001, 0.005, 0.01}; embedding dimension d $\in$ {128, 256, 512}; batchSize $\in$ {128, 256, 512} and L2 regularization $\in$ {0.001, 0.003, 0.005}. After experiments, the best parameter in CN15k is {lr = 0.001; d = 512; batchSize = 256; L2 = 0.003}, the best parameter in NL27k is{lr = 0.001; d = 512; batchSize = 256; L2 = 0.003}, and the best parameter in PPI5k is {lr = 0.001; d = 128; batchSize = 256; L2 = 0.003}.

C. BENCHMARK MODELS AND EXPERIMENTS

We chose TransE, DistMult, ConvKB, CapsE, RotatE and ComplEx [25] in the deterministic KG embedding models. We also choose $UKGE_{rect}$ and $UKGE_{logi}$ in the uncertain graph embedding models as benchmarks. $UKGE_{rect}$ and $UKGE_{logi}$ are two variants of UKGE,and the main difference between them is that $UKGE_{logi}$ uses the sigmoid activation function for output.

The discrimination ability of SUKE's evaluator and the accuracy of SUKE's confidence generator determine the quality of the results in the link prediction task. We performed confidence prediction, triple evaluation and fact classification tasks on the three datasets. The triple evaluation task is used to verify the effectiveness of SUKE's evaluator. Confidence prediction and fact classification tasks are used to verify the accuracy of SUKE's confidence generator.

D. CONFIDENCE PREDICTION

1) EVALUATION PROCESS

The confidence prediction task is defined as follows: Predict the missing confidence of a quadruple (h, r, t, ?). Deterministic models use energy scores as the confidence of the triples. This study uses SUKE's confidence generator to obtain the confidence of the triples. The evaluation indicators are the MSE and MAE. A good model should have low MAE and MSE.

2) RESULT ANALYSIS

The confidence prediction results are shown in Table 2.

TABLE 2. Confidence prediction results.

	CN15k		NL27k		PPI5k	
Metrics	MSE	MAE	MSE	MAE	MSE	MAE
ConvKB	25.05	42.57	13.3	24.28	37.26	57.09
CapsE	<u>6.55</u>	19.87	14.31	33.46	20.41	5.65
RotatE	13.3	29.6	8.4	19.7	34.82	55.10
URGE	10.32	22.72	7.48	11.35	1.44	6.00
$UKGE_{rect}$	8.61	<u>19.90</u>	<u>2.36</u>	<u>6.90</u>	<u>0.95</u>	<u>3.79</u>
$UKGE_{logi}$	9.86	20.74	3.43	7.93	0.96	4.07
SUKE	**5.12**	**17.82**	**0.77**	**3.19**	**0.29**	**1.95**

Bold font indicates the optimal result, and underline indicates the sub-optimal result.

As shown in Table 2, compared with $UKGE_{rect}$, SUKE has a significant improvement in MSE and MAE indicators, and has the smallest MAE and MSE. URGE only captures the neighbor information of entities, which cannot accurately model the rich relationships between the entities, resulting in an inaccurate confidence prediction. Although the results of $UKGE_{rect}$ and $UKGE_{logi}$ are more accurate than those of URGE, these models add negative triples and unknown facts during training, introducing unnecessary noise and thus affecting performance. In addition, it can be noticed that deterministic models have poor predictability. Since the energy score of deterministic models cannot be used as the probability value of the triple, it leads to higher MAE and MSE. For example, in the PPI5K data set, the average energy score of ConvKB is 0.98, while the true confidence average of the test set is 0.41. The confidence generator of the SUKE model only uses the confidence of positive triples during training. Therefore, the learning process can fit better, and the prediction results are more accurate.

E. TRIPLE EVALUATION TASK

1) EVALUATION PROCESS

The task of triple evaluation is to evaluate the rationality of triples. Consistent with UKGE's assessment method, we focus on the tail entity prediction. For an incomplete triple (h, r, ?), we take all entities in the KG as candidates for tail entities and calculate their rankings. In the evaluation task, this study uses SUKE's evaluator to conduct experiments and the evaluator has two scoring fusion methods: the linear fusion method is denoted as $SUKE_{line}$, and the multiplication fusion method is denoted as $SUKE_{mult}$.

NDCG is a common indicator for evaluating the quality of weighted sorted lists. It is the ratio of the actual sorted list gain to the expected sorted list gain, and its value range is between 0 and 1. In this study, the true confidence of the triples will be used as the gain, and the expected ranking is obtained by descending order of confidence. In the actual sorting results, if the triples with high confidence are ranked higher, the calculated NDCG will be higher. Therefore, a good model should have a high NDCG. This paper uses the average NDCG as the evaluation index and reports two versions, namely linear gain and exponential gain.

2) RESULT ANALYSIS

The triple evaluation results are shown in Table 3.

TABLE 3. Triple evaluation results.

	CN15k		NL27k		PPI5k	
Metrics	Linear	Exp	Linear	Exp	Linear	Exp
△TransE	0.601	0.591	0.730	0.722	0.710	0.700
△DistMult	0.689	0.677	0.911	0.897	0.894	0.880
△ComplEx	0.723	0.712	0.921	0.913	0.896	0.881
*ConvKB	0.740	0.727	0.901	0.884	0.909	0.894
*CapsE	0.778	0.763	0.930	0.916	0.951	0.932
*RotatE	0.771	0.776	0.818	0.816	0.961	0.952
△URGE	0.572	0.570	0.593	0.593	0.726	0.723
$\triangle UKGE_{rect}$	0.773	0.775	0.939	0.942	0.946	0.946
$\triangle UKGE_{logi}$	0.789	0.788	<u>0.955</u>	<u>0.956</u>	0.970	0.969
$SUKE_{mult}$	<u>0.807</u>	<u>0.801</u>	0.943	0.943	<u>0.984</u>	<u>0.983</u>
$SUKE_{line}$	**0.808**	**0.805**	**0.962**	**0.962**	**0.988**	**0.988**

Linear is the linear gain NDCG, Exp is the exponential gain NDCG, △ indicates that the result is from [11], * indicates the experimental results of our research, bold font indicates the optimal result, and underline indicates the suboptimal result.

Table 3 shows that SUKE gives the best performance. In the deterministic KG embedding methods, TransE has difficulties in handling complex relationships. Although DistMult's performance is slightly better than TransE, the model itself also has defects: when predicting, the (h, r, t) and (t, r, h) scores are the same. ComplEx embeds entities and relationships in the complex space and has stronger expressivity, so it performs best in deterministic embedding models. ConvKB and CapsE are models based on neural networks. Although their interpretability is not strong, they have achieved good performance in the evaluation task. RotatE performs best in deterministic models because it can model various relational patterns. However, deterministic models ignore the rich semantic information contained in the uncertainty, and when using the NDCG evaluation method, the performance is slightly worse. Among the methods of embedding UKG, URGE does not model complex relations well. The performance of $UKGE_{logi}$ is better than that of $UKGE_{rect}$, which because $UKGE_{logi}$ uses the sigmoid function to enhance the discrimination ability. UKGE relies on the confidence of both positive and negative triples during training, which may lead to the positive quadruples with low confidence, limiting the prediction performance. The model in this study introduces structure and uncertain scores for quadruples, which improves the ability to distinguish between positive and negative examples.

Table 3 shows that the linear fusion method has more advantages over the multiplicative fusion method. This is because the latter makes the uncertain and structural scores correlated. If a quadruple has high structural and low uncertain scores, the result may be small after multiplication; therefore, the effect is slightly worse. However, compared with existing models, the multiplicative fusion method is still very competitive.

3) EVALUATOR EFFECTIVENESS

From the experimental results in Section V-E-2, we can see that SUKE can distinguish positive and negative examples well, and $SUKE_{line}$ is superior to $SUKE_{mult}$. To verify the effectiveness of the evaluator, this study conducts experiments on the PPI5k dataset. For each triple of the test set, we randomly replace the tail entity to obtain a negative triple and determine the score of each triple through the evaluator. The evaluation results are shown in Fig. 3.

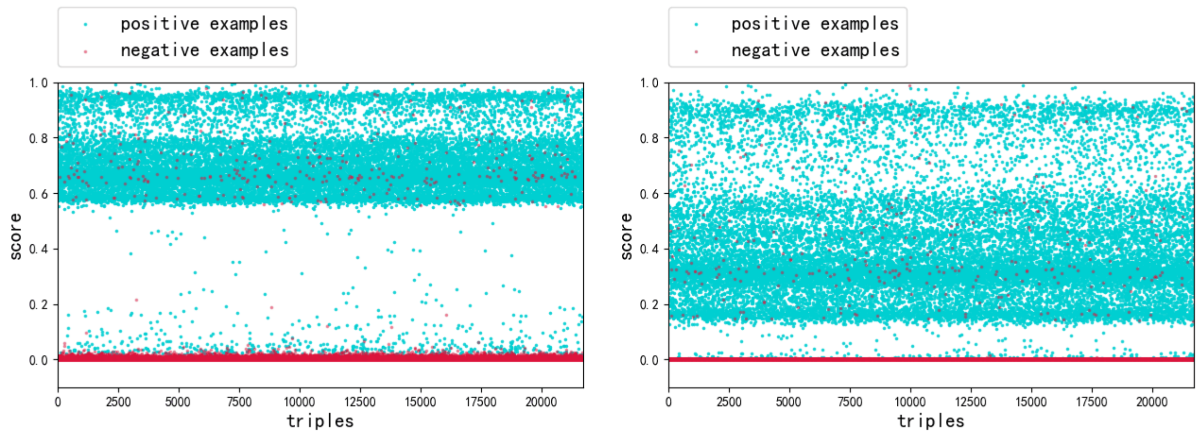
Fig. 3 shows that the evaluator puts on good classification performance, the scores for positive triples are significantly higher than those for negative triples, and the latter tend to be 0 in most cases. However, although $SUKE_{line}$ and $SUKE_{mult}$ produce similar evaluation performance in PPI5k, there is a big difference in their ability to distinguish. Compared with the linear fusion method, the difference between positive and negative examples with the multiplicative fusion method is relatively small, and the capability for classification fault tolerance is poor. This reason may also explain why $SUKE_{line}$ performs better. Exploring more reasonable fusion methods may help improve the performance of the evaluator.

F. FACT CLASSIFICATION

1) EVALUATION PROCESS

Consistent with the practice of Chen *et al.*, for each UKG, we define a specific threshold τ. If the confidence of the triple in the test set is higher than the KG specific threshold τ, then the triple is considered correct. This study uses SUKE's confidence generator to complete the classification task.

The triples in the test set are divided into two categories according to corresponding thresholds τ. The triples with confidence greater than or equal to τ contribute to the true class; otherwise, the false class. This study sets CN15k and



(a) (b)

FIGURE 3. Evaluation results of different fusion methods. The abscissa is the sequence of triples, and the ordinate is the evaluator score. The blue dots represent positive examples, and the red dots represent negative examples. a) Evaluation results of the linear fusion method; b) evaluation results of the multiplicative fusion method.

TABLE 4. Fact classification result.

	CN15k		NL27k		PPI5k	
Metrics	F-1	ACC	F-1	ACC	F-1	ACC
△TransE	23.4	67.9	65.1	53.4	83.2	98.5
△DistMult	<u>27.9</u>	71.1	72.1	70.1	86.9	97.1
△ComplEx	18.9	73.2	63.3	53.4	83.2	98.9
*ConvKB	22.4	70.6	77.7	64.7	86.7	96.7
*CapsE	18.6	77.4	79.1	65.5	82.2	94.3
*RotatE	24.8	75.1	86.5	80.1	90.4	99.1
△URGE	21.2	86.0	83.6	88.7	85.2	98.6
$\triangle UKGE_{rect}$	**28.8**	**90.4**	<u>92.3</u>	**95.2**	<u>95.1</u>	<u>99.4</u>
$\triangle UKGE_{logi}$	25.9	<u>90.1</u>	88.4	93.0	94.5	99.2
SUKE	26.2	81.0	**95.0**	<u>93.6</u>	**97.8**	**99.7**

△ indicates that the result is from [11], * indicates the experimental results of our research, bold font indicates the optimal result, and underline indicates the suboptimal result.

NL27k to $\tau = 0.85$, and PPI5k to $\tau = 0.80$. Under this setting, 20.4% fact triples in CN15k, 20.1% fact triples in NL27k, and 20.2% fact triples in PPI5k are considered correct.

F-1 score and accuracy rate (ACC) are selected as metrics. ACC is an intuitive classification index, which is the ratio of the number of correctly classified samples to the total number of samples. A good classifier should have a high ACC, but a classifier with a high ACC is not necessarily the best. The F-1 score is a comprehensive index of recall and precision. When F1 is higher, the classification is more effective.

2) RESULT ANALYSIS

The fact classification results are shown in Table 4.

The results show that although deterministic models can still take on the classification task, their capabilities are limited, and the existing deterministic models put on slightly worse performance than uncertain knowledge embedding models (UKGE and SUKE). Among the uncertain embedding models, URGE cannot model rich relations and therefore cannot miss the best performance. The UKGE model introduces negative examples and unknown facts during training, introducing unnecessary noise. Although the SUKE does not achieve the best performance on the CN15k, the F-1 score of the SUKE is close to the rest of the models. In addition, SUKE puts in the best classification performance on the NL27k and PPI5k datasets.

VI. CONCLUSION

For the knowledge representation problem of UKG, this paper proposes the SUKE model. SUKE includes two components, an evaluator and a confidence generator. The evaluator appraises triples' rationality based on the structure and uncertain information with two different score fusion methods. Considering that unknown facts are implied in the KG, this paper introduces probabilistic soft logic to obtain them and proposes a combination approach with unknown facts during the evaluator training. The confidence generator is used to obtain the confidence of each triple. The experimental results show that, compared with deterministic KG embedding models and other uncertain embedding models, SUKE shows great advantages in the confidence prediction, link prediction, and fact triple classification tasks.

Compared to other embedded models, SUKE model learns two vector representations for each entity and relationship, which undoubtedly increases the model complexity and difficulty of model training. A reasonable mapping function may solve this problem. In addition, we will explore more score fusion methods.

REFERENCES

[1] K. D. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: A collaboratively created graph database for structuring human knowledge," in *Proc. ACM SIGMOD Int. Conf. Manage. Data (SIGMOD)*, Vancouver, BC, Canada, Jun. 2008, pp. 1247–1250.

[2] G. A. Miller, "WordNet: A lexical database for English," *Commun. ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[3] A. Bordes, J. Weston, and N. Usunier, "Open question answering with weakly supervised embedding models," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Berlin, Germany: Springer, 2014, pp. 165–180.

[4] R. Hoffmann, C. Zhang, X. Ling, L. Zettlemoyer, and D. S. Weld, "Knowledge-based weak supervision for information extraction of overlapping relations," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 1. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 541–550.

[5] R. Speer, J. Chin, and C. Havasi, "Conceptnet 5.5: An open multilingual graph of general knowledge," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 1–9.

[6] T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, B. Yang, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, and J. Krishnamurthy, "Never-ending learning," *Commun. ACM*, vol. 61, no. 5, pp. 103–115, 2018.

[7] W. Wu, H. Li, H. Wang, and K. Q. Zhu, "Probase: A probabilistic taxonomy for text understanding," in *Proc. Int. Conf. Manage. Data (SIGMOD)*, 2012, pp. 481–492.

[8] I. Dellal, S. Jean, A. Hadjali, B. Chardin, and M. Baron, "Query answering over uncertain RDF knowledge bases: Explain and obviate unsuccessful query results," *Knowl. Inf. Syst.*, vol. 61, no. 3, pp. 1633–1665, Dec. 2019.

[9] M. Fan, Q. Zhou, and T. F. Zheng, "Learning embedding representations for knowledge inference on imperfect and incomplete repositories," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell. (WI)*, Oct. 2016, pp. 42–48.

[10] J. Hu, R. Cheng, Z. Huang, Y. Fang, and S. Luo, "On embedding uncertain graphs," in *Proc. ACM Conf. Inf. Knowl. Manage.*, Nov. 2017, pp. 157–166.

[11] X. Chen, M. Chen, W. Shi, Y. Sun, and C. Zaniolo, "Embedding uncertain knowledge graphs," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 3363–3370.

[12] D. Szklarczyk, J. H. Morris, H. Cook, M. Kuhn, S. Wyder, M. Simonovic, A. Santos, N. T. Doncheva, A. Roth, P. Bork, L. J. Jensen, and C. von Mering, "The STRING database in 2017: Quality-controlled protein–protein association networks, made broadly accessible," *Nucleic Acids Res.*, vol. 45, no. D1, pp. D362–D368, Jan. 2017.

[13] M. Nickel, V. Tresp, and H.-P. Kriegel, "A three-way model for collective learning on multi-relational data," in *Proc. ICML*, vol. 11, 2011, pp. 809–816.

[14] B. Yang, W.-T. Yih, X. He, J. Gao, and L. Deng, "Embedding entities and relations for learning and inference in knowledge bases," 2014, *arXiv:1412.6575*. [Online]. Available: http://arxiv.org/abs/1412.6575

[15] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2787–2795.

[16] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes," in *Proc. 28th AAAI Conf. Artif. Intell.*, 2014, pp. 1112–1119.

[17] G. Ji, S. He, L. Xu, K. Liu, and J. Zhao, "Knowledge graph embedding via dynamic mapping matrix," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics, 7th Int. Joint Conf. Natural Lang. Process.*, vol. 1, 2015, pp. 687–696.

[18] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," in *Proc. 29th AAAI Conf. Artif. Intell.*, Austin, TX, USA, Jan. 2015, pp. 2181–2187. [Online]. Available: http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9571

[19] Z. Sun, Z.-H. Deng, J.-Y. Nie, and J. Tang, "RotatE: Knowledge graph embedding by relational rotation in complex space," 2019, *arXiv:1902.10197*. [Online]. Available: http://arxiv.org/abs/1902.10197

[20] T. Dettmers, P. Minervini, P. Stenetorp, and S. Riedel, "Convolutional 2D knowledge graph embeddings," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 1–9.

[21] D. Q. Nguyen, T. D. Nguyen, D. Q. Nguyen, and D. Phung, "A novel embedding model for knowledge base completion based on convolutional neural network," 2017, *arXiv:1712.02121*. [Online]. Available: http://arxiv.org/abs/1712.02121

[22] D. Q. Nguyen, T. Vu, T. D. Nguyen, D. Q. Nguyen, and D. Phung, "A capsule network-based embedding model for knowledge graph completion and search personalization," in *Proc. Conf. North*, 2019, pp. 2180–2189.

[23] R. Xie, Z. Liu, F. Lin, and L. Lin, "Does william shakespeare REALLY write hamlet? Knowledge representation learning with confidence," 2017, *arXiv:1705.03202*. [Online]. Available: http://arxiv.org/abs/1705.03202

[24] Y. Shan, C. Bu, X. Liu, S. Ji, and L. Li, "Confidence-aware negative sampling method for noisy knowledge graph embedding," in *Proc. IEEE Int. Conf. Big Knowl. (ICBK)*, Nov. 2018, pp. 33–40.

[25] T. Trouillon, J. Welbl, S. Riedel, E. Gaussier, and G. Bouchard, "Complex embeddings for simple link prediction," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2016, pp. 1–10.

JINGBIN WANG received the M.S. degree in computer science from Fuzhou University, Fuzhou, China, in 2003. She is currently an Associate Professor with the College of Mathematics and Computer Science and the College of Software, Fuzhou University. Her current research interests include knowledge graph, relation reasoning, distributed data management, and knowledge representation. She has published over 20 articles in these areas.

KUAN NIE received the B.Sc. degree in engineering from Shanxi University, Shanxi, China, in 2018. He is currently pursuing the master's degree with Fuzhou University. His research interests include knowledge graph, knowledge representation, and intelligent technology.

XINYUAN CHEN received the M.S. degree in computer technology from Fuzhou University, Fuzhou, China, in 2018. He is currently a Lecturer with Fuzhou Melbourne Polytechnic. His research interests include natural language processing, knowledge representation, and inference.

JING LEI received the B.Sc. degree in computer technology from Fuzhou University, Fuzhou, China, in 2019, where she is currently pursuing the master's degree. Her research interests include knowledge graph, knowledge representation, and intelligent technology.