Knowledge Graph Embedding Based on Hyperplane and Quantitative Credibility

Shuo Chen[1], Lin Qiao[1], Biqi Liu[1], Jue Bo[1], Yuanning Cui[2](✉), and Jing Li[2]

[1] State Grid Liaoning Electric Power Supply Co, LTD., Shenyang 110004, Liaoning, China

[2] College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, Jiangsu, China

yuanningcui@163.com

Abstract. Knowledge representation learning is one of the research hotspots in the field of knowledge graph in recent years. How to improve the training algorithm of knowledge representation model and improve the accuracy of knowledge graph knowledge completion prediction is the main research goal in this field. Applying more implicit semantic information to model training is the primary means of improving accuracy. The traditional method does not consider the change of knowledge validity with time. So for this problem, we study the distribution law of quantitative changes of knowledge, design the model to simulate the quantitative changes of knowledge, put forward the concept of quantitative credibility, and apply it to the training algorithm of the model, and put forward A new learning method of knowledge representation QCHyTE. We compare the trained model with the best-recognized algorithms, and the results show that our improved algorithm greatly improves the prediction accuracy of the model.

Keywords: Knowledge graph · Time aware · Representation learning · Link Prediction

1 Introduction

The study of knowledge graphs has made great progress in recent years. Knowledge graph is a generalized formal description framework of semantic knowledge. It uses nodes to represent semantic symbols and edges to represent semantic relations between symbols. The common way is to store entities in the real world in the form of structured triples. And the relationship between entities. The way in which knowledge is expressed determines its expressive power and the complexity of semantic computing. The numerical representation method of the trigram's expression form knowledge graph has strong expressive ability, but it is difficult to use computer to do semantic calculation. It means that learning is a numerical representation method, which improves the semantic graph semantic calculation validness.

X. B. Zhai et al. (Eds.): MLICOM 2019, LNICST 294, pp. 583–594, 2019.
https://doi.org/10.1007/978-3-030-32388-2_50

Representation learning is one of the hot topics of knowledge graph research. It uses discrete symbols (entities, attributes, relationships, values, etc.) in the knowledge graph to represent continuous semantic values. This representation can reflect the semantic information of entities and relationships. Efficient computation of entities, relationships, and their complex semantic associations. At present, the knowledge graph indicates that there are two main categories of learning: one is the representation learning method based on tensor decomposition, and the most representative ones are RASCAL [1], DistMult [2], HoIT [3] and TuckER [4]. The idea of this type of method is to use tensor to represent the knowledge in the knowledge graph, and to complete the knowledge based on tensor decomposition, and achieved good results. The other type is translation-based representation learning methods, the most representative of which are TransE [5], TransH [6], TransR [7], TransD [8] and so on. They treat the relationship in the triple as a translation vector from the head to the tail. The goal of the training is to make the translation of the head entity vector through the relation vector close to the vector of the tail entity. The advantage of the tensor decomposition based method is that the information of the entire knowledge graph is integrated in the process of encoding entities and relationships. Its disadvantage is that the tensor is large for the Large Scale knowledge graph, and the decomposition process is computationally intensive; The translation-based model overcomes the problem of low learning efficiency can quickly complete representation learning in a large-scale knowledge graph. The disadvantage is that only a part of the information in the knowledge graph can be learned, and the expression power is not as good as the method based on tensor decomposition.

Time information is used in knowledge graph embedded learning, which is a new research direction in recent years. Most models assume that the knowledge graph exists in the same space-time condition, which is obviously not true. For example, Trump is the president of the United States. This fact was established in 2018 and was not established in 2014. Therefore, time information is an important implicit semantic information of the knowledge graph. Jiang [9] and others used the temporal information for the first time in the knowledge graph embedded learning, and achieved good results. Dasgupta [10] used the TransH [6] method to use time information as the main measure to distinguish one-to-many relationship, and proposed the HyTE [10] model. Their work graphs entities and relationships through temporal information to the temporal hyperplane and then computes the energy function. This method creatively embeds temporal information directly into the hyperplane space. Their experimental results show that the HyTE [10] model has better prediction effects than TransE [5], TransH [6], t-TransE [9] and HoIE [3].

From TransE to HyTE, it can be found from the development of the translation model that it is helpful to include more implicit semantic information in the model to improve the knowledge completion effect. But the more information that is not included, the better. Generally speaking, the more semantic information is included in the model, the more parameters need to be trained. The more parameters that need to be trained, the longer it takes to train the model. So the model needs to find a balance between expressiveness and efficiency.

We found that for datasets with time stamps, the duration of most relationships satisfies certain rules. For example, the relationship of GraduatedFrom, the valid duration is mostly from three years to six years, but a small part is from one years to two years or from seven years to ten years. We believe that this law is a priori knowledge, and the HyTE model cannot learn this knowledge duration the learning process. So we studied the law of the law of the quantitative change of knowledge, modeled the quantitative change, and used it to improve the model training algorithm.

The main contributions of our method includes:

(1) We propose the concept of quantitative credibility of knowledge. The concept of knowledge quantitative credibility is put forward, and its calculation formula is proposed by quantitative change modeling for meta-facts.
(2) An improved model training algorithm is proposed. The QCHyTE model was proposed by improving the HyTE model with quantitative credibility.
(3) Our experimental results demonstrate the validness of our approach. Design comparison experiments demonstrate the impact of our improvements on the accuracy of model predictions and analyze the causes of the impact.

2 Background

Before introducing the method of this paper, we will introduce TransE and TransH, and then introduce the temporal perception model HyTE in detail, because our method is based on the temporal perception model HyTE, and TransE and TransH are the basis of HyTE.

2.1 Knowledge Graph

The knowledge graph stores the entities in the real world and the relationships between the entities in the form of structured triples, expressed as $G = \{\varepsilon, R, \delta\}$, where $\varepsilon = \{e_1, e_2, \ldots, e_{|\varepsilon|}\}$ Representing a set of entities, $R = \{r_1, r_2, \ldots r_{|R|}\}$ represents a set of relationships, and $\delta \in R \times \varepsilon \times \varepsilon$ represents a set of triples in the knowledge graph.

2.2 TransE

TransE [5] uses a low-dimensional vector to represent each entity and relationship in the entity and relationship set of the knowledge graph, and uses the triples in the triple set as training samples. h, r, t in a triple (h, r, t) represent the head entity, relationship, and tail entity, respectively. TransE regards the relation vector as the translation vector from the head entity to the tail entity. For the two entity vectors $e_h, e_r \in R^n$, the difference between $e_h + e_r$ and e_t is used to score the translation effect. Its evaluation function can be expressed as: $f(h, r, t) = ||e_h + e_r - e_t||_{l_1/l_2}$. Where $||.||_{l_1/l_2}$ is l_1 norm or l_2-norm.

2.3 TransH

TransE has a poor predictive effect on one-to-many and many-to-one relationships. TransH [6] for this problem, each of the relations in the set R is represented by a hyperplane, and the unit normal vector of the hyperplane is denoted as w_r. Before the evaluation function is calculated, the head and tail entities are graphped to the relational hyperplane. The evaluation function of the TransH model can be expressed as:

$$f_\tau(h, r, t) = ||(e_h - w_r^T e_h w_r) + d_r - (e_t - w_r^t e_t w_r)||_2^2 \quad (1)$$

d_r represents the corresponding relationship between the pair of entities. For the same relationship, there can be multiple d_r, so TransH can better represent one-to-many, many-to-one, and many-to-many relationships than TransE.

2.4 HyTE

In some large knowledge graph, some meta facts are time stamped. These meta-facts can be structurally represented as $(h, r, t, [\tau_s, \tau_e])$, and $[\tau_s, \tau_e]$ represents the valid time of this triple.

The HyTE [10] model is a model designed on the basis of TransE [5] and inspired by TransH [6]. For these meta-facts containing event markers, Dasgupta et al. think that time is the main factor in their relationship between one-to-many and many-to-one. They use a hyperplane for each time, and the normal vector of hyperplane is recorded as w_t. Before calculating the evaluation function, the head and tail entities and relationship vectors are graphped to the temporal hyperplane, and then the evaluation function is calculated. The evaluation function of HyTE is:

$$f_\tau(h, r, t) = ||P_\tau(e_h) + P_\tau(e_r) - P_\tau(e_t)||_{l_{1/2}} \quad (2)$$

where $P_\tau(e_h) = e_h - w_\tau^T e_h w_\tau$, $P_\tau(e_t) = e_t - w_\tau^T e_t w_\tau$ and $P_\tau(e_r) = e_r - w_\tau^T e_r w_\tau$.

Where $P_\tau(?)$ represents the vector on the temporal hyperplane obtained by projecting the head entity, tail entity or relationship vector onto the time Label τ.

The main contribution of HyTE is that it splits the triple valid time into time labels, then uses the normal vector of the time hyperplane as the training parameter, and trains the entity with the relationship vector. The advantage of this is that the valid time information is subtly included in the model, so its prediction of the temporal perception of the knowledge graph is better than TransE and TransH.

3 Proposed Method: QCHyTE

Qualitative changes of events are caused by quantitative changes. The quantitative process of similar events is similar, and the time from the beginning to the qualitative change is also similar. Inspired by this rule, we modeled the quantitative process of similar events in the knowledge graph by studying the duration of events. QCHyTE is proposed by applying the learned rule of event quantity change to the improvement of knowledge representation model.

3.1 Quantitative Change Modeling

In general, a fact triple in a knowledge graph that contains a time stamp is subject to the process of invalid → valid → invalid in the time dimension. HyTE believes this process is abrupt. In other words, they believe that the establishment of knowledge at a certain point in time is a binary problem, not 0 or 1.

This is obviously contrary to our perception. A life from birth to death, an event from start to finish, is a gradual change, from quantitative change to qualitative change. So we believe that not all relationships are mutated, they undergo some random quantitative changes before they produce a qualitative change. We think this is the reason why many facts validly last for a certain distribution.

We extracted data from two relationships in the Wikidata dataset. The duration of their statistics is shown in the left part of Fig. 1. Their distribution satisfies the density function that is concentrated on a certain value and whose distribution pattern is close to the Gaussian distribution. In addition, some relationships occur instantaneously and end, and their start time and end time are the same. As shown in the right part of Fig. 1 below.

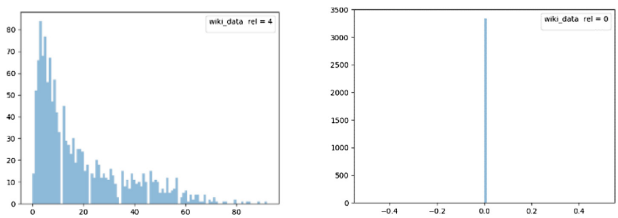

Fig. 1. The distribution of persistent relationships and transient relationship

Firstly, we divide the relationship into two categories according to the duration of the event, one is a persistent relationship, and the second is a transient relationship. The duration distribution of the relationships shown in Fig. 1 is the subject of our main study.

Persistent Relationships. A persistent relationship is a relationship in which the relationship is valid for a certain period of time, such as LiveIn and WorkFor. Since this type of relationship is not zero, we can think of it as a process of quantitative change.

We propose the quantitative credibility (QC) to represent the probability of an event, which represents the degree of credibility of the meta facts in the knowledge graph that persists at a certain point in time. It has been observed that the beginning and end of an event often satisfy the Gaussian distribution, so we use the difference between the two Gaussian distribution functions to simulate the distribution of QC (Fig. 3).

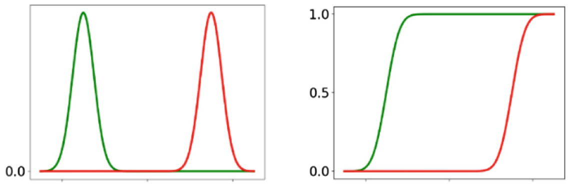

Fig. 2. The PDF and CDF of the start and end of the valid duration

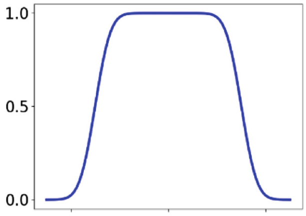

Fig. 3. The curve of quantitative credibility

As shown in Fig. 2, we use two Gaussian distribution density functions to simulate the probability that the event starts and ends near the two time points τs and τe. The two graphs are the distribution functions of the two Gaussian distributions. We use the distribution function starting with the valid time. The distribution function at the end of the deduction valid time is also shown in Fig. 2. Therefore the calculation function of quantitative credibility is:

$$\mathrm{QC}(\tau_s, \tau_e, \tau_p) = \int_{-\infty}^{\tau_p} \frac{1}{\sigma\sqrt{2\pi}} e - \frac{(y-\tau_s)^2}{2\sigma^2} dy - \int_{-\infty}^{\tau_p} \frac{1}{\sigma\sqrt{2\pi}} e - \frac{(y-\tau_e)^2}{2\sigma^2} dy \quad (3)$$

In this way, we have well simulated the quantitative credibility of a triple in the knowledge graph in the time dimension, which provides the basis for us to use quantitative change credibility for knowledge embedded representation.

Transient Relationship. The transient relationship means that this relationship occurs at a certain time and does not persist, such as WasBornIn and DeadIn. Since this type of relationship has a duration of 0, we can think of it as a process of mutation. Its quantitative credibility is 1 at the valid time point and 0 at the invalid time point.

3.2 Model Training Based on Quantitative Credibility

In the HyTE model, the distribution of the triples in the knowledge graph in the time dimension is regarded as binary. They thought that the QC at all time points between the τs and τe of the triples is 1. Since HyTE is also a kind of model based on the energy function, the quantitative change credibility represents the distribution of energy in some ways. However, the objective fact is that the stability at these points in time is different. We observe that the credibility at the intermediate time point of the event establishment period is higher than the edge time point. So we tried to add QC to the scoring function and use this implicit information for KG Embedding duration training.

Both TransE and models based on it use a training strategy that uses negative sampling to speed up training. The QC of positive and negative samples will be calculated differently. The QC of a positive sample typically experiences a change from 0 to almost 1 then to 0 in the time dimension. The negative sample is an unknown random triple, which is not in the correct triple. It is generally considered to be invalid in any time plane duration the training process, so the QC of the negative sample is 1. QC can be expressed as:

$$\mathrm{QC}\left(\tau_s, \tau_e, \tau_p, \mathrm{r}\right) = \begin{cases} \int_{-\infty}^{\tau_p} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(y-\tau_s)^2}{2\sigma^2}} - \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(y-\tau_e)^2}{2\sigma^2}} dy + q, \ triple \in \mathcal{D}^+ \ and \ r \in R_p \\ 1, \ triple \in \mathcal{D}^- \ or \ r \notin R_p \end{cases} \tag{4}$$

τ_p in the formula represents the current time point, the value of the positive sample QC is in (0, 1), and the QC of the negative sample is 1. Since our QC does not consider the data outside the marked valid period in the knowledge graph, therefore, the training evaluation function of the positive sample is actually reduced. If such QC is directly applied to the instructional training, the role of the positive sample in training will be reduced. So for a positive sample, we add q on the basis of QC, the purpose is to correct the QC of the positive sample to maintain the balance between the positive and negative samples.

QCHyTE, like HyTE, is a time-driven model. As with the HyTE model, our model graphs the evaluation function to the time plane as well. We use time to express it in a hyperplane. For T time points, we represent the normal vectors of T different time hyperplanes. The evaluation function we propose can be expressed as:

$$f_T\left(h, r, t, \tau_s, \tau_e, \tau_p\right) = \mathrm{QC}\left(\tau_s, \tau_e, \tau_p, \mathrm{r}\right) ||(e_h + e_r - e_t) - \omega_\tau^T(e_h + e_r - e_t)\omega_\tau||_{l_{1/2}} \tag{5}$$

Compared with HyTE, QCHyTE does not have more parameters, so when calculating the loss function, it can be calculated by the following function:

$$\mathrm{L} = \sum_{\tau \in [T]} \sum_{x \in \mathcal{D}_\tau^+} \sum_{y \in \mathcal{D}_\tau^-} \max(0, f_\tau(x) - f_\tau(y) + \gamma) \tag{6}$$

Where D_τ^+ represents a set of valid triples at each time point τ, and a negative sample is collected on the basis of the valid triples to obtain a set D_τ^- of negative samples. The method of taking a negative sample is a negative sampling method that does not count time:

$$\mathcal{D}^- = \{(h', r, t, \tau) | h' \in \xi, (h', r, t) \notin \mathcal{D}^+\} \cup \{(h, r, t', \tau) | t' \in \xi, (h, r, t') \notin \mathcal{D}^+\} \tag{7}$$

4 Experimental Results

In this part, we will test our model through experiments. We evaluate the model with the effect and accuracy of Link Prediction, which is a common evaluation method in the KG Embedding field. Then we compare the experimental results with the current advanced methods and analyze the reasons for this result.

4.1 Datasets

WikiData and YAGO are two large knowledge graphs, and most of the KG Embedding method experiments use these two datasets. YAGO11k and Wikidata12k are tense-aware knowledge graph datasets composed of meta-facts containing event marker information from these two datasets. The meta facts in YAGO11k are stored in the form (#factID, occurSince, ts), (#factID, occurUntil, te), which is extracted from the fact that the YAGO data set contains time stamps, which contains 20.5k triples, 10623 entities and 10 frequent relationships. Similar to YAGO11k, Wikidata12k is extracted from the Wikidata dataset and contains 24 frequent relationships, 40k triples, and 12.5k entities. This is the two data sets used in HyTE.

In order to better prove the validity of the model training algorithm proposed in this paper, we further extract the data according to the persistence and relational type of the data set relationship, and extract 15525 pieces of data containing the continuous relationship in the triple from the YAGO data set, and then the facts. The entities and relationships contained in the data are extracted to form a new continuous relational data set YGP10K. Sampling the same extraction method for Wikidata data, we have (Table 1).

Table 1. Dataset

Dataset	#R	#E	#Train	#Test	#Valid
Wikidata12K	24	12554	32497	4062	4062
YAGO11K	10	10623	16408	2051	2046

4.2 Details

Evaluation Indicators. In order to accurately evaluate our model approach, we used a general evaluation index for the TransE-based model. For each triple that needs to be tested, we remove the head and tail entities separately, and then replace them with all the entities in the dataset. After graphping to the time hyperplane, we use the evaluation function to score and sort all the entity evaluation results. The average ranking of the correct entities in all entities (Mean Rank) is used as an evaluation indicator, and then the percentage of the top ten data of all entities in the correct entity (Hit@10) is counted as an evaluation criterion. Similarly, the relationship vector in the triple is deleted, then replaced by all the relationships in the dataset, and scored using the evaluation function. The average ranking of the correct relationship in all entities (Mean Rank) is used as an evaluation indicator, and then the percentage of the data with the correct relationship among the first in all relationships (Hit@1) is used as an evaluation index.

Baseline Settings. The first method is TransE, which is the most classic method based on the translation model. It does not consider time stamp information, uses a triple set as a training set, and outputs a vector for each entity and each relationship. HoIE is a KG representation learning method. Its prediction effect is state-of-the-art. It is also a method that does not consider time stamping. TransH is a method based on TransE. It is the first time to apply hyperplane to knowledge representation learning. The HyTE method is also inspired by this. t-TransE is also a translation-based model that applies time information to knowledge representation learning for the first time, but this method does not have direct training in time learning. HyTE is an improved model method based on TransE and TransH. It considers time as the main factor in generating one-to-many and many-to-one relationships. Our method is based on HyTE. QCHyTE is our proposed method. We add the distribution of events in time as prior knowledge to the training. For the specific introduction, please see Sect. 3.
We used the baseline method to test on our dataset. The learning process of TransE, TransH and HoIE did not use time information. The learning process of t-TransE, HyTE and QCHyTE used time information. We set the appropriate parameters for these algorithms.

Parameter Setting. For all methods, we keep b = 10k on both datasets, the value of the embedded dimension is chosen in {64, 128, 256}, the boundary is chosen in {1, 2, 5, 10}, the learning rate is {0.01, Selected from 0.001, 0.0001}, the adjustment factor is selected in {0.3, 0.4, 0.5, 0.6, 0.7}. In the course of the experiment, we observed a dimension of 128, a boundary of 10, a learning rate of 0.0001, an adjustment factor of 0.6, and an evaluation model using the 1 paradigm to obtain the best model.

4.3 Results and Analysis

Entity Prediction. On the two datasets we prepared, we made the predictions of the head and tail entities respectively. The indicators tested were the Mean Rank and Hits@10 (%) of the correct head and tail entities. Comparing our experimental results with the experimental results of the baseline method, the results are as follows (Table 2).

Table 2. Results of entity prediction

Dataset	Wikidata12K				YAGO11K			
Metric	Mean rank		Hits@10 (%)		Mean rank		Hits@10 (%)	
	Tail	Head	Tail	Head	Tail	Head	Tail	Head
HoIE [3]	734	808	25.0	12.3	1828	1953	29.4	13.7
TransE [5]	520	740	11.0	6.0	504	2020	4.4	1.2
TransH [6]	423	648	23.7	11.8	354	1808	5.8	1.5
t-TransE [9]	283	413	24.5	14.5	292	1692	6.2	1.3
HyTE [10]	179	237	41.6	25.0	**107**	1069	38.4	16.0
QCHyTE	**111**	**175**	**61.3**	**38.1**	115	**783**	**39.6**	**20.4**

On the Wikidata12K dataset, our method has achieved very good results, and each evaluation index has been significantly improved compared with the baseline method. On the YAGO11K, except for the MR of the tail, which is worse than the HyTE, other prediction effect indicators are Got an improvement. We analyzed the data of YAGO11K and Wikidata12K and found that the distribution of events is more regular in Wikidata12K. There are a large number of transient triples in the YAGO11K dataset, and there are a large number of missing values, and the duration of the relationship is relatively short. This is the reason why tail prediction is worse. We further compared the results of HyTE and QCHyTE predictions for tail and found that even if MR increases, QCHyTE has an improved prediction of 46% of the data, and 38% of the data predicts a decrease, and the prediction results are good. Triples tend to be better, and the test triples, which were poorly predicted, will be even worse. This is also the reason why Hit@10 (%) does not fall back when MR increases. Below we provide a comparison of the predicted results of several HyTE and QCHyTE tail entities (Table 3).

Table 3. Comparation of entity prediction between HyTE and QCHyTE

Test quadruples	HyTE_tail_pred	QCHyTE_tail_pred
James_Baker, isAffiliatedTo,?, [1970,####]	Republican_Party_(United_States), Democratic_Party_(United_States), Unionist_Party_(United_States), Independent_politician	Democratic_Party_(United_States), Republican_Party_(United_States), Unionist_Party_(United_States), Walsall_F.C.
Esperanza_Baur, isMarriedTo,?[1946-1954]	Esperanza_Baur, Josephine_Wayne, John_Wayne,	Esperanza_Baur, John_Wayne, Josephine_Wayne,
Edith_Baumann_(politician), isAffiliatedTo,? [1946-1973]	Communist_Party_of_Germany, Socialist_Unity_Party_of_Germany, Social_Democratic_Party_of_Germany, Bulgarian_Communist_Party	Socialist_Unity_Party_of_Germany, Communist_Party_of_Germany, Nazi_Party, Oxford

Relation Prediction. On the two datasets we prepared, we made a prediction of the relationship. The indicators tested were Mean Rank and Hits@1 (%) of the correct relationship vector. Comparing our experimental results with the experimental results of the baseline method, the results are as follows (Table 4).

Table 4. Results of relation prediction

Dataset	Wikidata12K		YAGO11K	
Metric	Mean rank	Hits@1 (%)	Mean rank	Hits@1 (%)
HolE [3]	2.23	83.96	2.57	69.3
TransE [5]	1.35	88.4	1.7	78.4
TransH [6]	1.4	88.1	1.53	76.1
t-TransE [9]	1.97	74.2	1.66	75.5
HyTE [10]	1.13	92.6	1.23	81.2
QCHyTE	**1.10**	**94.2**	**1.14**	**86.1**

For the prediction of relationships, our method improves the accuracy of the prediction. We analyze the test triples with improved prediction results. The results are shown in the following Table 5.

Table 5. Comparation of relation prediction between HyTE and QCHyTE

Test quadruples	HyTE_rel_pred	QCHyTE_rel_pred
Norman_Borlaug, ?, University_of_Minnesota, [1937-1942]	isMarriedTo, graduatedFrom	graduatedFrom, wasBornIn
Francisco_Gallardo,?, Puskás_Akadémia_FC,[2013-2014]	isMarriedTo, playsFor	playsFor, isMarriedTo
Konstantinos_Tsatsos,?, Independent_politician, [1967-1974]	isMarriedTo, isAffiliatedTo	isAffiliatedTo, isMarriedTo
Donovan_Leitch_(actor),?, London,[1946-1967]	diedIn, wasBornIn	wasBornIn, diedIn
Kate_O'Mara,?, Leicester,[1939-1939]	isMarriedTo, wasBornIn	wasBornIn, diedIn

5 Conclusions

In order to add the distribution law of the quantitative change for of the meta-facts to KG Embedding, we proposed QCHyTE. We study the distribution of meta-facts in time and propose the credibility of QC quantitative change. This is semantic information implicit in the time dimension, and we apply it as a priori knowledge to KG Embedding. Through experiments, we compared the prediction effects of QCHyTE and BaseLine methods and verified the validness of QCHyTE. Then compared with the training time of HyTE, it proves that the training efficiency of QCHyTE has also been improved. In the course of the experiment, we found that the time stamp of the knowledge graph has many rules that can be applied to training, such as the periodicity of duration, which is the direction of our future work. In addition, knowledge representation learning is also an important basis for knowledge fusion and text extraction. How to use the time-sensing knowledge representation learning for knowledge fusion and text extraction is also worth studying.

Acknowledgement. The research was supported by the State Grid Liaoning Electric Power Supply CO., LTD, and we are grateful for the financial support for the "Key Technology and Application Research of the Self-Service Grid Big Data Governance (SGLNXT00YJJ S1800110)".

References

1. Nickel, M., Tresp, V., Kriegel, H.-P.: A three-way model for collective learning on multi-relational data. In: Proceedings of the 28th International Conference on Machine Learning, ICML 2011, pp. 809–816. Omnipress (2011)
2. Yang, B., Yih, W.-T., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases. arXiv preprint arXiv:1412.6575 (2014)

3. Nickel, M., Rosasco, L., Poggio, T.: Holographic embeddings of knowledge graphs. In: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI 2016, pp. 1955–1961. AAAI Press (2016)
4. Balažević, I., Allen, C., Hospedales, T.M.: TuckER: tensor factorization for knowledge graph completion (2019)
5. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multirelational data. In: Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, vol. 26, pp. 2787–2795. Curran Associates, Inc. (2013)
6. Wang, Z., Zhang, J., Feng, J., Chen, Z.: Knowledge graph embedding by translating on hyperplanes. In: Proceedings of the Twenty Eighth AAAI Conference on Artificial Intelligence, AAAI 2014, pp. 1112–1119. AAAI Press (2014)
7. Lin, Y., Liu, Z., Sun, M., Liu, Y., Zhu, X.: Learning entity and relation embeddings for knowledge graph completion. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, AAAI 2015, pp. 2181–2187. AAAI Press (2015)
8. Ji, G., He, S., Xu, L., Liu, K., Zhao, J.: Knowledge graph embedding via dynamic graphping matrix. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), vol. 1, pp. 687–696 (2015)
9. Jiang, T., et al.: Encoding temporal information for time-aware link prediction. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 2350–2354. Association for Computational Linguistics (2016)
10. Dasgupta, S.S., Ray, S.N., Talukdar, P.: HyTE: hyperplane-based temporally aware knowledge graph embedding, pp. 2001–2011 (2018). D18-1225