A Knowledge Graph Completion Method for Telecom Metadata Based on the Spherical Coordinate System

KAICHENG ZHANG1, HAN WANG2, MINGCHUAN YANG1, XINCHI LI1, XIAOQING XIA1, AND ZHIXIA GUO1
1Institute of Big Data and Artificial Intelligence, China Telecom Corporation Limited Beijing Research Institute, Beijing 102209, China
2School of Information Science and Technology, Fudan University, Shanghai 200433, China
Corresponding author: Kaicheng Zhang (zhangkc1@chinatelecom.cn)

ABSTRACT In the telecommunications field, the problem of data island is caused by the separation and isolation of data. They are distributed in different systems including the business support system (BSS), management support system (MSS), and operation support system (OSS). The common idea is to use global ID mapping to break data barriers. However, using the direct global ID mapping of raw data has the problems of large data scale and the inability to guarantee privacy and security. With this in mind, constructing and completing a metadata knowledge graph to fabric the data is a feasible approach. Considering the particularity of the telecom metadata knowledge graph and the need for hierarchical distinction in business and semantic abstraction, we propose a deep learning method and framework based on the spherical coordinate system. It can be extended to a poly-spherical coordinate system and add a pre-training process composed of word2vec and a clusterer. Experimental results show that our method achieves state-of-the-art on our dataset, MDCT and has excellent performance on two public datasets, FB15k-237 and WN18RR.

INDEX TERMS Spherical coordinate system, telecom metadata, knowledge graph completion.

I. INTRODUCTION
In the telecommunications field, data comes from different systems like business support system (BSS), management support system (MSS) and operation support system (OSS). It also comes from different provinces and cities that have their own unique systems. These systems are not connected to each other and do not follow a unified standard and specification, then a data island is formed. To solve this problem, a common idea is to build a data lake or data warehouse to store all the required data in one place. However, it will have the following problems: data leakage in data transmission, cost of hardware and software required, and poor scalability. Therefore, we need to find a way to avoid data transmission. A possible way is to construct a metadata knowledge graph to connect data from different systems. In this way, the above-mentioned problems could be mitigated and eliminated.

The knowledge graph was first proposed in 2012 by Google as a special variant of the semantic web. Reference [1] It is composed of the schema layer and the data layer logically. The data layer uses a basic cell of (entity, relation, entity) and the properties of entities to describe the concepts and interconnections of the physical world. We can symbolize a triple as follows: \((h, r, t)\), with \(h\) denoting the head entity, \(t\) denoting the tail entity and \(r\) denoting the relation from \(h\) to \(t\).

There are two routes for building a knowledge graph: top-down and bottom-up. A general knowledge graph used to adopt the former; a domain-specific knowledge graph, which can also be viewed as an explicit conceptualization to a high-level subject-matter domain [2], used to adopt the latter. Whatever which route, the technical architecture of a knowledge graph consists of the following aspects: information acquisition that includes entity extraction, relation extraction,
and attribute extraction, knowledge confusion that includes conference resolution and entity disambiguation, and knowledge inference that includes simple inference such as knowledge graph completion and complex inference like multi-hop inference.

In all these processes, knowledge graph completion is one of the most important for the following reasons: 1) it can connect different entities so that it can achieve the continuous expansion of the knowledge graph; 2) it is the basis of the complex inference such as multi-hop inference.

The Knowledge Graph is essentially a product of symbolism, which relies heavily on expert experience and mathematical logic. With Lecun et al. [3], [4], and [5] proposing and improving the convolutional neural network (CNN), including the further development and improvement of the generative models such as GAN [6], [7], [8] and contrastive learning models [9], [10], [11], connectionism [12] has become mainstream in the field of artificial intelligence research. Therefore, it’s an important direction that how to use the idea of connectionism to study the knowledge graph, including building a knowledge graph, mining the information between entities, and doing the knowledge reasoning. The proposal of knowledge representation [13] learning would be a good solution.

Knowledge representation learning refers to obtaining a dense, low-dimensional, and real-valued vector for each entity and relation by adopting a model. Structured embedding (SE) [14] firstly projects the head entity and tail entity into the space of relation \( r \) by using two parameter matrices. However, the SE has high complexity and lacks synergy. The latent factor model (LFM) [15], [16] provides a bilinear transformation matrix \( M_r \), which has good synergy and can effectively describe the semantic connection between entities. Dismult [17] further simplifies the transformation matrix \( M_r \) to a diagonal matrix. It could reduce not only the model complexity but also improve the model effect. Mikolov et al. [18] found the translation invariance of word vectors by using Word2vec [19] in 2013. Inspired by this discovery, Bordes et al. [20] proposed the TransE model, which regards the relationship in the knowledge graph as a certain translation vector between the head entity and tail entity. Compared with the previous model, the TranE model can directly build complex semantic connections between entities and relations and perform amazingly on large-scale sparse knowledge graphs with fewer model parameters and low computational complexity. However, TransE has a poor expression effect for complex relations, so TransH [21] and TransR [22] introduce hyperplane and multi-relational semantic space methods, respectively, to improve the expression effect of complex relations. TransD [23] and TranSparse [24] have further optimized heterogeneity and imbalance of entities and relationships. TransG [25] and KG2E [26] introduce Gaussian distribution to represent entities and relationships, respectively. TransA [27] changes the distance metric of the loss function from \( L1 \) or \( L2 \) norm to Mahalanobis Distance.

All of the above methods are based on the Cartesian coordinate system, and they all ignore a very important problem, that is, the entities have different levels of semantic abstraction. The methods of RotatE [28] and HAKE [29] introduce complex space and the polar coordinate system, respectively, to solve this problem. For example, in the triple of \( (Obama, is, man) \), \( Obama \) refers to a specific individual, who is the 44th president of the United States, while \( man \) refers to each adult person who is male. Obviously, the entity \( Obama \) has a lower level of semantic abstraction than the entity \( man \). Therefore, encoding entities with different levels of semantic abstraction equally and ignoring their semantic levels will lose lots of information. Besides, the metadata of china telecom also has the need to distinguish service levels. Therefore, we propose a method based on the spherical coordinate system to construct and complete the telecom metadata knowledge graph.

In summary, we summarize our work into the following three points:

1) We propose a basic model framework based on the idea of the spherical coordinate system, where the radial distance denotes the level of semantic and business abstraction, and the polar angle and azimuth angle are used to distinguish entities with the same level.

2) In order to enhance the expressive power, we also further extend our spherical coordinate system to the poly-spherical coordinate system.

3) Besides, we also introduce a pre-training process to change the traditional way of using one-hot encoding or random initialization as input. Our model achieves state-of-art on both metadata datasets of china telecom and public datasets, and our model can also be used as an infrastructure to extend expressions for different problems and needs.

II. RELATED WORK

In this section, we will introduce some classical methods and analyze their advantages and disadvantages in chronological order of appearance.

Structured embedding (SE) [14] is a primary method for learning structure embeddings of knowledge graph. It mapped all entities to the D-dimensional space, and two projection matrices, \( M_{r,1} \) and \( M_{r,2} \) are defined to project the head and tail entities in the triplet. The loss function of the model is defined as follows:

\[
f_{SE}(h, t) = |M_{r,1}h - M_{r,2}t|_{L1}
\]  

(1)

In the process of SE model link prediction, the relation corresponding to the relation matrix of entity vectors closest to each other is found according to the following formula:

\[
\text{arg min} |M_{r,1}h - M_{r,2}t|_{L1}
\]  

(2)

SE model has a simple structure and does not fully consider the semantic relationship between entities. Different projection matrices are used for the head and tail entities, leading
to poor coordination. Compared with SE model, The latent factor model (LFM) [15] uses the redefined and decomposed loss function to solve the data coefficient problem. The loss function is defined as follows:

\[
 f_{\text{LFM}}(h, t) = \| \mathbf{h}^T \mathbf{M}_h \mathbf{I}_t \|_2
\]

LFM occupies a small space and memory, has strong generalization ability, and can well cope with sparse data. It has good scalability and flexibility but poor interpretability and cannot consider the correlation degree of context. Dismult [17], on the basis of the LFM model, further restricts the relation matrix, which is a diagonal matrix and applies the embedding-based method to mine rules.

With the word2vec [18] proposed in 2013, Mikolov et al. found that there is a translation invariance phenomenon in the word vector space. Inspired by this, Bordes et al. proposed the TransE [20]. In the D-dimension space of the word vector space, a relation is regarded as the translation from head entity to tail entity, and a correct triplet should satisfy that the head entity vector plus relation vector is approximately equal to the tail entity vector. In contrast, the wrong triplet does not meet the above conditions, so the loss function is as follows:

\[
 f_{\text{TransE}}(h, t) = \| \mathbf{h} + \mathbf{r} - \mathbf{t} \|_1
\]

The TransE model is simple and has a small amount of computation, which supports processing a large amount of data. However, the model is too simple and cannot deal with complex knowledge networks well.

On the basis of the TransE, the TransH [21] is proposed to improve the ability dealing with complex relationships. Different projection matrices are defined for different relationships, and each relationship defines a hyperplane so that the same entity is in different hyperplanes in different relationships. The loss function is defined as follows:

\[
 f_{\text{TransH}}(h, t) = \| \mathbf{h}^T \mathbf{M}_h \mathbf{I}_t \|_1
\]

Compared with the TransE, the TransH has overcome the limitations of the expression for complex relationships to a certain degree and made different relationship representation. However, relations and entities are still in the same semantic space, and the TransH is not enough to express more complex relationships. The TransR [22] maps entities and relations into different semantic spaces based on the TransE. Relations are no longer represented in a single semantic space but in multiple relational spaces, and the translation of head entity and tail entity is also completed in the corresponding relational space. The loss function is defined as follows:

\[
 f_{\text{TransR}}(h, t) = \| \mathbf{h}^T \mathbf{M}_h \mathbf{I}_t \|_1
\]

Compared with the previous model, TransR has significantly improved the effect and can better distinguish entity attributes and relational semantics. However, TransR model adopts the same projection matrix at the head and tail of the entity, which lacks interaction with the relationship. Moreover, the introduction of spatial projection leads to an increase in parameters, time, and computational complexity. Based on the TransR, TransD [23] adopts different projection matrices for the head and tail entities, and connects the head and tail entities and their relations through projection matrices. The loss function of the model is defined as follows:

\[
 f_{\text{TransD}}(h, t) = \| \mathbf{h}^T \mathbf{M}_h \mathbf{I}_t \|_1 \| \mathbf{I}_h^T \mathbf{M}_h \mathbf{I}_t \|_2
\]

The TransD also considers the diversity of entities and relationships and has fewer parameters and no matrix or vector multiplication operation, which improves the computing power of the model. However, the design of the model loss function is too simple, and each dimension has equal status, which cannot effectively distinguish the importance of different dimensions.

In the KG2E [26], the knowledge graph is represented in a multi-dimension Gaussian space. Each entity and relationship is represented by a Gaussian distribution. The mean value represents its position, and the variance represents its certainty. The SGD optimizer was used to update the parameters. Two loss functions were set in the KG2E model, which was KL divergence under asymmetric condition and expectation under symmetric condition, respectively. KL divergence under asymmetric conditions:

\[
 E(h, r, t) = E(P_e, P_r) = D_{KL}(P_e, P_r)
\]

\[
 = \int_{x \in \mathbb{R}^k} N(x; \mu_e, \Sigma_e) \log \frac{N(x; \mu_e, \Sigma_e)}{N(x; \mu_r, \Sigma_r)} dx
\]

\[
 = \frac{1}{2} \left\{ \text{tr} \left( \Sigma_e^{-1} \Sigma_r \right) + (\mu_e - \mu_r)^T \Sigma_r^{-1} (\mu_e - \mu_r) - \log \det(\Sigma_r) + k_e \right\}
\]

The expectation in symmetric case:

\[
 E(h, r, t) = \log E(P_e, P_r) = \log N(0; \mu_e - \mu_r, \Sigma_e + \Sigma_r)
\]

\[
 = \frac{1}{2} \left\{ (\mu_e - \mu_r)^T (\Sigma_e + \Sigma_r)^{-1} (\mu_e - \mu_r) + \log \det(\Sigma_e + \Sigma_r) + k_e \log(2\pi) \right\}
\]

KG2E model performs well in modeling the uncertainty of entities and relationships in the knowledge graph and performs better in link prediction and triplet classification in the data set than other methods. However, the model does not explicitly consider the influence of different relationships and does not classify entity types more carefully.

Considering the heterogeneity of relationships, RotatE [28] extends entities and relations to the complex vector space. It defines each relationship as the rotation from the source
entity to the target entity, redefines the loss function, and proposes the self-adversarial sampling loss function as follows:

\[ L_{\text{RotatE}} = - \log \sigma (y - f_i(h_i, t)) \]

\[ - \sum_{i=1}^{n} p(h_i', r_i, t_i) \log \sigma (f_i(h_i', t_i) - y) \]

\[ \text{while } p((h_i', r_i, t_i) | (h, r, t)) = \frac{\exp f_i(h_i', t_i)}{\sum \exp f_i(h_i', t_i)} \]

The model is flexible and can infer and model various complex relations, especially symmetry/anti-symmetry, inversion, and composition. Compared with the previous model, the effect is significantly improved, and the self-adversarial sampling loss function can reduce the amount of calculation in the optimization process. On the basis of this model, HAKE [29], InterHT [30], PairRE [31], NodePiece [32] and TranS [33] are proposed to improve in various aspects.

With the the proposal of Graph Neural Network (GNN) models [34], including Graph Convolutional Neural Network (GCN) [35], GraphSage [36] and so on, GNN is regarded as an effective method to learn the structure of the graph by aggregating the information of neighbor nodes. So it is obvious that it is applied to knowledge graph completion. Zhang et al. [37] did a lot of analysis and ablation experiments to rethink how to use GCN in knowledge graph completion. With Ashish Vaswani et al. [38] finding attention mechanism and proposing the structure of the Transformer, graph attention network (GAT) [39] also came into being. Wang et al. [40] uses the GAT to learn entity-level aggregation and relation-level aggregation.

### III. METHOD

Quantitative approaches have been employed since Bordes et al. proposed TransE. They independently make great progress on learning complex relationships, ambiguous uncertainty of relationships, others, etc. However, a major problem with this kind of application is the different levels of the entity’s semantic abstraction. HAKE proposes a method based on the polar coordinate system to solve this problem. We find that there is also business abstractions in the field of metadata of telecom. However, the existing methods cannot solve or deal with this problem well. In view of this, we propose a translation framework based on the spherical coordinate system to do this. In this section, we will introduce the basic model framework and its extensions, including the following three parts: translation based on the spherical coordinate system, translation based on the polyspherical coordinate system, and pre-training process using word2vec and a clusterer.

### A. TRANSLATION MODEL BASED ON THE SPHERICAL COORDINATE SYSTEM

In this part, we creatively propose a method based on the spherical coordinate system, aiming at the different levels of the entity’s semantic and business abstraction. As Figure 1(b) illustrates, for each entity in the triplet \((h, r, t)\), we use a 3-tuple \((\rho, \theta, \varphi)\) to denote, where \(\rho\), the radical distance, denoting the semantic and business level of entity’s abstraction. The higher the abstraction degree of the entity, the shorter the value of \(\rho\). For entities with the same value of \(\rho\), we use the polar angle \(\theta\) and azimuth angle \(\varphi\) to distinguish them. In this way, we model all entities on the spherical coordinate system, which can effectively represent the level of semantic and business abstraction. In our method, each entity has only three dimensions, \(\rho\), \(\theta\), and \(\varphi\), which greatly reduce the complexity. We define the score function of each dimension as follows:

\[
\left\{ \begin{array}{l}
\rho_{\text{score}} = \left\| h_\rho + r_\rho - t_\rho \right\|_{L_2} \\
\theta_{\text{score}} = \frac{\sin(h_\theta + r_\theta - t_\theta)}{4} \\
\varphi_{\text{score}} = \frac{\sin(h_\varphi + r_\varphi - t_\varphi)}{8}
\end{array} \right.
\]

where \(0 \leq \theta \leq \pi, 0 \leq \varphi < 2\pi\), and \(L_2\) denotes the Euclidean norm.

In view of above, we can define the overall score function as follows:

\[
f_i(h, t) = \rho_{\text{score}} + \alpha \ast \theta_{\text{score}} + \beta \ast \varphi_{\text{score}}
\]

\[
\begin{align*}
&= \left\| h_\rho + r_\rho - t_\rho \right\|_{L_2} + \alpha \ast \left\| \frac{h_\theta + r_\theta - t_\theta}{4} \right\|_{L_2} \\
&\quad + \beta \ast \left\| \frac{h_\varphi + r_\varphi - t_\varphi}{8} \right\|_{L_2}
\end{align*}
\]

where \(\alpha, \beta\) denote the weights of each dimension.

During the training process, existing knowledge graphs only have positive triples like \((h, r, t) \in S\). We need to sample some negative triples such as \((h', r, t') \in S’\) to train with positive samples together. Therefore, the entire loss function is as follows:

\[
L = \sum_{(h, r, t) \in S} \sum_{(h', r, t') \in S'} -\log \sigma(r - f_i(h, t)) - \log \sigma(f_i(h', t') - r)
\]
Then we expand Equation 15 and write it as follows:

\[
\begin{aligned}
\theta_1 &= \sin\left(\frac{h_{\theta_1} + r_{\theta_1} - t_{\theta_1}}{4}\right) L_2 \\
\theta_2 &= \sin\left(\frac{h_{\theta_2} + r_{\theta_2} - t_{\theta_2}}{4}\right) L_2 \\
&\cdots \\
\theta_n &= \sin\left(\frac{h_{\theta_n} + r_{\theta_n} - t_{\theta_n}}{4}\right) L_2
\end{aligned}
\tag{14}
\]

where \(0 \leq \theta_1, \theta_2, \ldots, \theta_n \leq \pi, 0 \leq \theta_n \leq 2\pi\), and \(L_2\) denotes the Euclidean norm.

In the same way as spherical coordinate system, then we can get the score function of polyspherical coordinate system as follows:

\[
f_r(h, t) = \rho_{\text{score}} + \alpha_1 \star \theta_1 + \alpha_2 \star \theta_2 + \cdots + \alpha_n \star \theta_n \tag{15}
\]

Then we expand Equation 15 and write it as follows:

\[
f_r(h, t) = \left\| h_{\rho} + r_{\rho} - t_{\rho} \right\|_{L_2} + \alpha_1 \star \left\| \sin\left(\frac{h_{\theta_1} + r_{\theta_1} - t_{\theta_1}}{4}\right) \right\|_{L_2}
\]
\[
+ \alpha_2 \star \left\| \sin\left(\frac{h_{\theta_2} + r_{\theta_2} - t_{\theta_2}}{4}\right) \right\|_{L_2} + \cdots \\
+ \alpha_n \star \left\| \sin\left(\frac{h_{\theta_n} + r_{\theta_n} - t_{\theta_n}}{4}\right) \right\|_{L_2}
\tag{16}
\]

where \(\alpha_1, \alpha_2, \ldots, \alpha_n\) denote the weights of each dimension.

### C. PRE-TRAINED BY WORD2VEC AND THE CLUSTERER

In knowledge representation learning, a common practice is to use one-hot encoding or random initialization as input. On the one hand, such input will lead to a too long training time or cannot achieve the best effect. On the other hand, the process of dividing the data set is carried out with triples as the smallest unit. This means that there will be a subset of entities that exist only in the triples of the test or validation set, but not in the triples of the training set. This phenomenon will have a significant impact on our experimental results, especially in the case of a large proportion of telecom metadata. And we will discuss and analyze it in Section IV-D.

Therefore, we propose a pre-training stage using the module of word2vec and a clusterer to implement it. As Figure 1(a) illustrates, before the training stage, we pass our data through the module of word2vec to get the embedding of each entity and then put it into the module of the clusterer. And in our method, we choose the K-Means algorithm as our clusterer, where the value of \(k\) is set as a hyper-parameter.

The only difference from the polyspherical coordinate system is that there is one more item in the score function as follows:

\[
c_{\text{score}} = \left\| h_c + r_c - t_c \right\|_{L_2} \tag{17}
\]

Then the score function can be defined as follows:

\[
f_r(h, t) = \rho_{\text{score}} + \alpha_1 \star \theta_1 + \alpha_2 \star \theta_2 + \cdots + \alpha_n \star \theta_n + \lambda \star c_{\text{score}} \tag{18}
\]

Expanding the Equation 18:

\[
f_r(h, t) = \left\| h_{\rho} + r_{\rho} - t_{\rho} \right\|_{L_2} + \lambda \star \left\| h_c + r_c - t_c \right\|_{L_2}
\]
\[
+ \alpha_1 \star \left\| \sin\left(\frac{h_{\theta_1} + r_{\theta_1} - t_{\theta_1}}{4}\right) \right\|_{L_2}
\]
\[
+ \alpha_2 \star \left\| \sin\left(\frac{h_{\theta_2} + r_{\theta_2} - t_{\theta_2}}{4}\right) \right\|_{L_2} + \cdots \\
+ \alpha_n \star \left\| \sin\left(\frac{h_{\theta_n} + r_{\theta_n} - t_{\theta_n}}{4}\right) \right\|_{L_2}
\tag{19}
\]

### IV. EXPERIMENTS

Our approach is evaluated on the following data sets: FB15k-237, WN18RR, and MDCT. The first two are public encyclopedia datasets, while the last dataset is a metadata
dataset in the telecommunications domain. Then in the following parts, we will introduce the data sets, experimental setup, and experimental results of link prediction. We will show how our method performs state-of-art against several methods in the three datasets mentioned above.

A. DATA SETS

1) MDCT
MDCT is a metadata dataset from China Telecommunication, including 15 kinds of services, 45 tables, and 2904 fields. In this dataset, the domain name, table name, and field name are entities, and we have marked six relationships of them through expert experience. In summary, MDCT is composed of 2175 entities, 4471 triples, and 6 kinds of relationships.

2) FB15k237
Metaweb company released the semantic database project Freebase online in March 2007. The entire Freebase data storage is a large graph, each point is defined by type/object, and edges are defined by type/link. Whether a model or a topic, its data is stored as a point in the graph database and linked by edges. There are 1.9 billion triples in the entire Freebase knowledge graph. FB15k237 is a subset of the Freebase, with 15k meaning that there are 15k subject terms in the knowledge base and 237 meaning that there are 237 kinds of relations.

3) WN18RR
WordNet is a database that describes the association characteristics between English words. It links English nouns, verbs, adjectives, and adverbs with synonyms linked to each other through semantic relationships to determine word definitions. WN18RR is a WordNet subset composed of 40943 entities and 11 kinds of relationships.

B. EXPERIMENTAL SETUP OF LINK PREDICTION

Link prediction means predicting the missing link of the triple \((h, r, t)\). Referring to the setting of TransE, we randomly mask \(h\) or \(t\) of a triple \((h, r, t)\) and use our model to get a predicted value. This predicted value is then used to calculate the similarity with other entities through cosine distance in the Euclidean space. Afterward, we select some candidates according to their similarities and sort the candidates. The higher the similarity is and the higher the ranking is, the higher the probability of being judged by the model as correct is. Finally, we evaluate the performance of a model based on whether or not the actual correct entity is in the candidates or the ranking of the candidates.

Common model evaluation indicators for knowledge graph completion are \(MR, MRR,\) and \(Hits@N\). The full name of \(MR\) is \(MeanRank\), meaning the average ranking of ground truth entities in all candidates. \(MRR,\) \(MeanReciprocalRanking\), is the reciprocal of each ranking and then averaged. \(Hits@N\) refers to selecting \(N\) out of all the prediction results. If a hit is recorded, that is to say, if the ground truth is in the \(N\) candidates, it is recorded as 1. On the contrary, it is recorded as 0. Finally, the probability value of the overall hit is calculated as \(Hits@N\). The larger the value of \(MRR\) and \(Hits@N\), the better the model effect is. Considering the scale of data and intuitiveness of the display effect, we choose \(MRR\) and \(Hits@10\) as evaluation indicators to compare the effect of our model with other models.

We generate a batch of wrong triples during the negative sampling process to train with the correct triples. However, in negative sampling, for the kinds of \(1 - n, n - 1,\) or \(n - n\) relationships, if we select one from other entities, there is a possibility that the correct triplet is regarded as the wrong triplet for training. To solve this problem, we use two modes, \(raw\) and \(filter\), to do the negative sampling. The mode of \(raw\) is to directly mask \(h\) or \(t\) of a triple \((h, r, t)\) and then select one other entity from all entities, and the mode of \(filter\) adds a step to ensure that the generated triple is different from any of the positive triples.

C. EXPERIMENTAL RESULTS

In our experiments, we choose TransE, TransH, TransR, Rotate, and Hake to compare with our method. As Table 2 and Figure 2 illustrates, our method achieves the best performance on both \(MRR\) and \(Hits@10\). Besides, as Table 1 illustrates, our method achieves the best performance on both metrics on two public datasets, FB15k-237 and WN18RR. Our method is not only suitable for the completion of the telecom metadata knowledge graphs with high professional knowledge requirements, but also can be used for the completion of public encyclopedic knowledge graphs.

D. DISCUSSION AND ANALYSIS

As we have pointed out in SectionIII-C, the process of dividing the data set has a result that there will be a subset of entities that exist only in the triples of the test or validation set, but
TABLE 2. Experimental results of link prediction on MDCT.

|         | MDCT(raw)  | MDCT(filter) |
|---------|------------|--------------|
|         | MRR | Hit@10 | MRR | Hit@10 |
| TransE  | 0.163 | 0.312   | 0.277631 | 0.439265 |
| TransH  | 0.162 | 0.316   | 0.278 | 0.438 |
| TransR  | 0.168 | 0.315   | 0.276 | 0.424 |
| Rotate  | 0.177 | 0.313   | 0.291 | 0.442 |
| HAKE    | 0.171 | 0.298   | 0.285 | 0.406 |
| Ourmodel| 0.177 | 0.334   | 0.307 | 0.412 |
| Ourmodel+word2vec+clusterer | 0.182 | 0.329 | 0.311 | 0.444 |

TABLE 3. Experimental results of link prediction on public datasets.

|         | WN18RR(raw) | WN18RR(filter) | FB15k237(raw) | FB15k237(filter) |
|---------|-------------|---------------|---------------|-----------------|
|         | MRR | Hit@10 | MRR | Hit@10 | MRR | Hit@10 | MRR | Hit@10 |
| TransE  | 0.196 | 0.326 | 0.196 | 0.326 | 0.183 | 0.303 | 0.288 | 0.475 |
| TransH  | 0.198 | 0.338 | 0.198 | 0.338 | 0.188 | 0.312 | 0.291 | 0.486 |
| TransR  | 0.203 | 0.331 | 0.203 | 0.331 | 0.192 | 0.323 | 0.296 | 0.506 |
| Rotate  | 0.201 | 0.352 | 0.201 | 0.352 | 0.182 | 0.307 | 0.271 | 0.470 |
| HAKE    | 0.199 | 0.319 | 0.199 | 0.319 | 0.179 | 0.300 | 0.275 | 0.445 |
| Ourmodel| 0.205 | 0.325 | 0.205 | 0.325 | 0.18 | 0.298 | 0.265 | 0.423 |
| Ourmodel+word2vec+clusterer | 0.207 | 0.336 | 0.207 | 0.336 | 0.191 | 0.301 | 0.294 | 0.468 |

FIGURE 3. The trend of performance indicators, MRR and Hits@N, under different conditions on three data sets, MDCT, FB15k237, and wn18rr, during the training process.

FIGURE 4. The data distribution of MDCT and FB15k237.

not in the triples of the training set. Such cases can be ignored when the proportion of such entities is small. However, when the number of such entities reaches a certain proportion, the impact on the accuracy cannot be ignored. As Figure3(b)(c) illustrate, we can find that on public datasets, FB15k-237 and WN18RR, the evaluation metrics MRR and Hits@N are almost indistinguishable whether the entities in the test set appear in the train set or not. However, as Figure3(a) illustrates, on metadata of china telecommunication, MDCT, the evaluation metric MRR and Hits@N have a clear difference. We infer that the reason for the difference between the MDCT and public datasets is the proportion of entities which only exist in the test set. As Figure4(a)4(b) shows that our inference is valid.

V. CONCLUSION AND FUTURE WORK

In conclusion, for the construction and completion of china telecom metadata knowledge graph, we propose a framework based on the spherical coordinate system. It uses the spherical radius to represent the semantic and business level of metadata. Moreover, it uses the polar angle and the azimuth angle to distinguish metadata that has the same semantic and business level. Further, to increase the model’s expressive power, we extend our model to the polyspherical coordinate system and add the pre-training process of word2vec and clusterer. Experimental results demonstrate that our method not only achieves state-of-the-art on metadata of china telecom but also both an excellent performance on public datasets, Fb15k-237 and WN18RR.

In future work, our approach will continue to improve in the following aspects:

1) We will further analyze complex relationships such as $1 - n$ and three types of relationships: symmetry/antisymmetry, inversion, and composition.
2) We only use the basic negative sampling algorithm for the negative sampling process. The different effects of raw and filter modes show how choosing the negative sampling method to generate Negative samples has become a critical factor in limiting the model’s performance. Furthermore, we will conduct continuous research and optimization in this part in the future.

3) Our method adopts the theory based on the spherical coordinates system to distinguish different semantics and business levels of entities. There are also problems like sparsity and imbalance in entities and relationships. Validating the effectiveness of our model on the above aspects and how to improve it is also a focus of our future research.

4) Our approach is to provide a knowledge representation learning framework for learning semantic and business abstraction levels. In the future work, we will consider adding the GNN and the Transformer to improve the effect.

REFERENCES

[1] J. Pujara, H. Miao, L. Getoor, and W. Cohen, “Knowledge graph identification,” in Proc. Int. Semantic Web Conf. Cham, Switzerland: Springer, 2013, pp. 542–557.

[2] B. Abu-Salah, “Domain-specific knowledge graphs: A survey,” J. Netw. Comput. Appl., vol. 185, Jul. 2021, Art. no. 103076.[Online]. Available: https://www.sciencedirect.com/science/article/pii/S1084804520300990

[3] Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel, “Handwritten digit recognition with a back-propagation network,” in Proc. Adv. Neural Inf. Process. Syst., 1990, pp. 394–400.

[4] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, “Backpropagation applied to handwritten zip code recognition,” Neural Comput., vol. 1, no. 4, pp. 541–551, 1989.

[5] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in Proc. Adv. Neural Inf. Process. Syst., vol. 27, 2014, pp. 139–144.

[7] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, S. Bengupta, and A. A. Bharath, “Generative adversarial networks: An overview,” IEEE Signal Process., vol. 35, no. 1, pp. 53–65, Jan. 2018.

[8] M. Mirza and S. Osindero, “Conditional generative adversarial nets,” 2014, arXiv:1411.1784.

[9] T. Wang and P. Isola, “Understanding contrastive representation learning through alignment and uniformity on the hypersphere,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 9929–9939.

[10] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, “Unsupervised feature learning via non-parametric instance discrimination,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 3733–3742.

[11] A. Van Den Oord, Y. Li, and O. Vinyals, “Representation learning with contrastive predictive coding,” 2018, arXiv:1807.03748.

[12] J. A. Fodor and Z. W. Pylyshyn, “Connectionism and cognitive architecture: A critical analysis,” Cognition, vol. 28, nos. 1–2, pp. 3–71, Mar. 1988.

[13] Q. Wang, Z. Mao, B. Wang, and L. Guo, “Knowledge graph embedding: A survey of approaches and applications,” IEEE Trans. Knowl. Data Eng., vol. 29, no. 12, pp. 2724–2743, Dec. 2017.

[14] A. Bordes, J. Weston, R. Collobert, and Y. Bengio, “Learning structured embeddings of knowledge bases,” in Proc. 35th AAAI Conf. Artif. Intell., 2011, pp. 1–6.

[15] R. Jenatton, N. Roux, A. Bordes, and G. R. Obozinski, “A latent factor model for highly multi-relational data,” in Proc. Adv. Neural Inf. Process. Syst., vol. 25, 2012, pp. 1–9.

[16] I. Sutskever, J. Tenenbaum, and R. R. Salakhutdinov, “Modelling relational data using Bayesian clustered tensor factorization,” in Proc. Adv. Neural Inf. Process. Syst., vol. 22, 2009, pp. 1–8.

[17] 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.

[18] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” in Proc. Adv. Neural Inf. Process. Syst., vol. 26, 2013, pp. 1–9.

[19] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” 2013, arXiv:1301.3781.

[20] 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., vol. 26, 2013, pp. 1–9.

[21] Z. Wang, J. Zhang, J. Feng, and Z. Chen, “Knowledge graph embedding by translating on hyperplanes,” in Proc. AAAI Conf. Artif. Intell., vol. 28, no. 1, 2014, pp. 1–9.

[22] 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., 2015, pp. 1–7.

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

[24] G. Ji, K. Liu, S. He, and J. Zhao, “Knowledge graph completion with adaptive sparse tensor matrix,” in Proc. 30th AAAI Conf. Artif. Intell., 2016, pp. 1–7.

[25] H. Xiao, M. Huang, Y. Hao, and X. Zhu, “TransG: A generative mixture model for knowledge graph embedding,” 2015, arXiv:1509.05488.

[26] S. He, K. Liu, G. Ji, and J. Zhao, “Learning to represent knowledge graphs with Gaussian embedding,” in Proc. 24th ACM Int. Conf. Inf. Knowl. Manage., Oct. 2015, pp. 623–632.

[27] H. Xiao, M. Huang, Y. Hao, and X. Zhu, “TransA: An adaptive approach for knowledge graph embedding,” 2015, arXiv:1509.05490.

[28] 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.

[29] Z. Zhang, J. Cai, Y. Zhang, and J. Wang, “Learning hierarchy-aware knowledge graph embeddings for link prediction,” in Proc. AAAI Conf. Artif. Intell., vol. 34, no. 3, 2020, pp. 3065–3072.

[30] B. Wang, Q. Meng, Z. Wang, D. Wu, W. Che, S. Wang, Z. Chen, and C. Liu, “InterHT: Knowledge graph embeddings by interaction between head and tail entities,” 2021, arXiv:2102.04897.

[31] L. Chao, J. He, T. Wang, and W. Chu, “PairRE: Knowledge graph embeddings via paired relation vectors,” 2020, arXiv:2011.03798.

[32] M. Galkin, E. Denis, J. Wu, and W. L. Hamilton, “NodePiece: Compositional and parameter-efficient representations of large knowledge graphs,” 2021, arXiv:2106.12144.

[33] X. Zhang, Q. Yang, and D. Xu, “TransS: Transition-based knowledge graph embedding with synthetic relation representation,” 2022, arXiv:2204.08401.

[34] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, “The graph neural network model,” IEEE Trans. Neural Netw., vol. 20, no. 1, pp. 61–80, Dec. 2009.

[35] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” 2016, arXiv:1609.02907.

[36] W. Hamilton, Z. Ying, and J. Leskovec, “Inductive representation learning on large graphs,” in Proc. Adv. Neural Inf. Process. Syst., vol. 30, 2017, pp. 1–11.

[37] Z. Zhang, J. Wang, J. Ye, and F. Wu, “Rethinking graph convolutional networks in knowledge graph completion,” in Proc. ACM Web Conf., Apr. 2022, pp. 798–807.

[38] I. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” in Proc. Adv. Neural Inf. Process. Syst., vol. 33, no. 8, pp. 3961–3973, Aug. 2020.

122677
KAICHEM ZHANG received the B.S. degree in control science and engineering from Shandong University, in 2018, and the M.S. degree in software engineering from Shandong University, in 2021. He is currently an Artificial Intelligence Engineer with the Institute of Big Data and Artificial Intelligence and the China Telecom Research Institute. His research interests include knowledge graph embedding, knowledge graph completion, and graph neural networks.

HAN WANG received the B.S. degree from the School of Information Engineering, Zhengzhou University, Zhengzhou, Henan, in 2021. She is currently pursuing the M.S. degree with the School of Information Science and Engineering, Fudan University, Shanghai, China. Her research interests include network security, communications encode, and deep learning.

MINGCHUAN YANG received the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, in 2002. He is currently the Director of the Institute of Big Data and Artificial Intelligence and the China Telecom Research Institute. His research interests include big data and artificial intelligence.

XINCHI LI received the graduate degree in computer application from the Computer Network Information Center, Chinese Academy of Sciences. She is a Senior Engineer. Currently, she is the Director of the Big Data Center of China, Telecom Research Institute. She has published more than ten papers and written two monographs. She is mainly engaged in technical research and product development in the fields of big data and blockchain.

XIAOQING XIA was born in 1994. She received the M.S. degree from the Beijing University of Technology, Beijing, China, in 2020. Her main research interests include big data and information security.

ZHIXIA GUO was born in Longhai, Zhangzhou, Fujian, China, in 1993. She received the M.S. degree in geographic information systems from Beijing Normal University, Beijing, China, in 2019. From 2019 to 2022, she was a Researcher with the Research Institute, China Telecom Corporation Ltd., Beijing. Her research interests include data mining and natural language processing.