

Article

Hyperbolic Directed Hypergraph-Based Reasoning for Multi-Hop KBQA

Guanchen Xiao ¹, Jinzhi Liao ¹, Zhen Tan ^{1,*}, Yiqi Yu ² and Bin Ge ¹¹ Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha 410073, China² People's Liberation Army, Guangzhou 510600, China

* Correspondence: tanzhen08a@nudt.edu.cn

Abstract: The target of the multi-hop knowledge base question-answering task is to find answers of some factoid questions by reasoning across multiple knowledge triples in the knowledge base. Most of the existing methods for multi-hop knowledge base question answering based on a general knowledge graph ignore the semantic relationship between each hop. However, modeling the knowledge base as a directed hypergraph has the problems of sparse incidence matrices and asymmetric Laplacian matrices. To make up for the deficiency, we propose a directed hypergraph convolutional network modeled on hyperbolic space, which can better deal with the sparse structure, and effectively adapt to the problem of an asymmetric incidence matrix of directed hypergraphs modeled on a knowledge base. We propose an interpretable KBQA model based on the hyperbolic directed hypergraph convolutional neural network named HDH-GCN which can update relation semantic information hop-by-hop and pays attention to different relations at different hops. The model can improve the accuracy of the multi-hop knowledge base question-answering task, and has application value in text question answering, human-computer interactions and other fields. Extensive experiments on benchmarks—PQL, MetaQA—demonstrate the effectiveness and universality of our HDH-GCN model, leading to state-of-the-art performance.



Citation: Xiao, G.; Liao, J.; Tan, Z.; Yu, Y.; Ge, B. Hyperbolic Directed Hypergraph-Based Reasoning for Multi-Hop KBQA. *Mathematics* **2022**, *10*, 3905. <https://doi.org/10.3390/math10203905>

Academic Editors: Di Jin, Liang Yang and Danilo Costarelli

Received: 6 September 2022

Accepted: 11 October 2022

Published: 21 October 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: hyperbolic space; directed hypergraph convolutional network; knowledge base QA; multi-hop reasoning

MSC: 68T07

1. Introduction

Knowledge base question answering (KBQA) has been a hot task in the field of natural language processing and is very challenging [1]. Several different QA datasets have been proposed, such as the Stanford Question Answering Dataset (SQuAD) [2,3], NarrativeQA [4] and CoQA [5], and the kind of reasoning based on these datasets is termed single-hop reasoning, since it requires reasoning only over a single piece of evidence [6]. For a QA task of single-hop reasoning, the performance of previous work [1,7] has been improved a lot over the last years.

However, in real-world QA tasks, obtaining answers often requires multi-hop reasoning [8], that is, to find a knowledge path consisting of multiple pieces of knowledge in the knowledge base to deduce the answer. Figure 1 shows a two-hop KBQA example. The reasoning path starts from the entity mentioned in the query and consists of the relations at each hop and the intermediate entities. The methods mentioned above that focus only on single-hop reasoning lack the ability to deal with multi-hop reasoning QA tasks. To solve a multi-hop QA task, some work has been proposed recently which can be mainly divided into two categories. One is the neural network-based methods such as models in [9,10] and the other is the graph neural network-based methods such as [11]; this has achieved desirable performance [12].

When we deal with multi-hop tasks, generally, entities extracted from the query begin to be retrieved in the knowledge base, go down to the next hop according to the specific relationship of different hops, repeat this step to form a reasoning path, and finally find the final answer. In this process, we argue that semantic relational information is crucial for multi-hop reasoning, while previous studies have not fully exploited semantic relational information. The work in [13] computes relation-specific transformations by separating different relations, which does not consider semantic relation information. Ref. [14] does not require updating relational information during multi-hop reasoning, but exploits relational information to obtain attention for static graphs.

Query: Which people starring in the 94th Oscar-winning film for Best Picture?

Reasoning Path: “*Child of Deaf Adults*” → *Winning_Oscar_Best_Picture* → *Starring_in* → ○

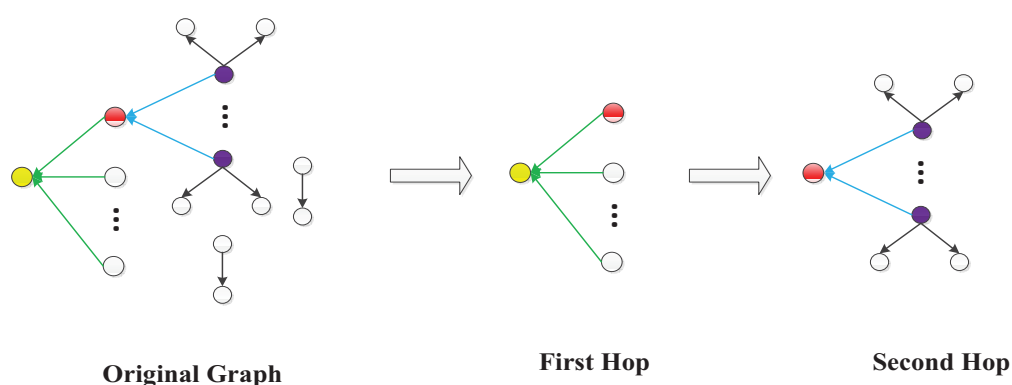


Figure 1. Sketch of an exemplar of multi-hop question answering based on knowledge graph. In the figure, arrows with different colors represent the relations in two hops that need to be extracted in order to obtain the reasoning path to the query, and colored dots represent the entities related to the question in knowledge base.

In addition, the pairwise connections between nodes based on a graph network (GNN) are insufficient with respect to fully representing the higher-order relationships between relationships and entities in the knowledge graph. Recently, some main works on hypergraph convolutional networks (HGNN) have been proposed. HGNN uses hyperedges to connect more than two nodes at the same time, which is conducive to imitating human reasoning and accurately locating a group of entities connected by the same relationship, rather than reasoning entity by entity. The disadvantage is that HGNN is aimed at undirected hypergraphs, while knowledge graphs are directed, and each triplet has a specific directional meaning. HGNN collects potential learning relationships from connected entities, but does not reveal and further utilize them.

Based on the study in the hypergraph neural network introduced above, a directed hypergraph convolutional network-based model for multi-hop KBQA (2HR-DR) was proposed [15]. 2HR-DR models the entities extracted from questions and their related relationships and entities in the knowledge base into directed hypergraphs, and then uses Directed Hypergraph Convolutional Networks (DHGCN) [15] to predict relations hop-by-hop and form a sequential relation path to make the reasoning interpretable. 2HR-DR can explicitly learn and update relation information and dynamically concentrate on different relations at different hops.

Although 2HR-DR can better solve some of the challenges mentioned above, it still has some disadvantages. First, using directed hypergraphs to model entities and relationships may lead to a situation in which an entity can be related to many entities based on some relations while it may be related to a few entities based on some other relations. For example, as shown in Figure 1, when constructing a hypergraph of the query, the number of entities (actors) related to “*Child_of_Deaf_Adults*” can be very small based on some relations such

as “Staring_in” while a large number of entities (actors) do “Act_in” this movie. This results in a large difference in the number of nodes contained by each hyperedge in the modeled directed hypergraph. In that case, the incidence matrices of constructed hypergraphs will be much sparser, and that will have a bad effect on the training efficiency and accuracy of the model. Second, 2HR-DR used the directed hypergraph convolution network, which needs the eigenvalue decomposition of Laplacian matrices when calculating the spectrum convolution of hypergraphs, and that requires that the Laplacian matrices are real symmetric matrices (we are not able to ensure that non-symmetric matrices can certainly perform eigenvalue decomposition). However, as for directed hypergraph convolution networks, since each hyperedge has a direction, the degree of each node is supposed to be divided into in degree and out degree, which are different in most cases; this leads to the fact that Laplacian matrices are often asymmetric matrices.

To solve the problems mentioned above, we firstly propose a Hyperbolic Directed Hypergraph Convolutional Network (HDH-GCN) for a directed hypergraph to take the direction of information transmission into account. We investigate hyperbolic embedding spaces [16] and manage to map the sparse data points and hypergraph to the hyperboloid manifold directly. The rationale is that hyperbolic space has a stronger ability than Euclidean space to accommodate networks with long-tailed distributions and sparse structures [17], which is also verified in our experiments. On that basis, we propose a Hyperbolic Directed Hypergraph Convolutional Network (HDH-GCN)-based framework for multi-hop QA. This framework explicitly updates the relation information and dynamically focuses on specific relations at every hop of the query. In addition, we record the semantic representation of the relationship in each hop, and the representation of the relationship in every hop is influenced by the representation of the relationship in the previous hops, which makes the QA task interpretable to a large extent.

In summary, we make the following contributions:

- For solving the problem of sparse incidence matrices of directed hypergraphs modeled on a knowledge base, we design a method of modeling a directed hypergraph in hyperbolic space.
- Based on the hyperbolic directed hypergraph, we propose a Hyperbolic Directed Hypergraph Convolutional Network (HDH-GCN) for a directed hypergraph and design a framework on this basis that can handle multi-hop knowledge base question-answering tasks well.
- The modules constitute a new model, namely, HDH-GCN for handling the multi-hop knowledge base question-answering task. Through the experiments on several real-world datasets, we confirm the superiority of HDH-GCN over state-of-the-art models.

2. Related Work

2.1. Multi-Hop Question Answering

The multi-hop KBQA model can be basically divided into two types. The first is to apply the neural network mentioned earlier. These models use the previous single-hop question-answering method [18–20] for multi-hop question-answering tasks. Xu et al. improved KVMemNet to achieve better results across multiple triples [10]. Zhong et al. used coarse-grained modules and fine-grained modules [21]. Part of the method introduces an end-to-end framework, which is explicitly designed to simulate the step-by-step reasoning process involved in multi-hop QA and MRC. Kundu et al.’s [22] model constructed a path connecting questions and candidate answers, and then scored them through the neural architecture. Jiang et al. [23] also constructed a proposer to propose an answer from each root to leaf path in the reasoning tree, extract a key sentence containing the proposed answer from each path and finally combine them to predict the final answer. However, these methods lack consideration of graph structure information.

The other kind of method is based on graph neural networks. Sun et al. learnt what to retrieve from the KB and corpus and then reasoned over the built graph [24]. Tu et al. employed GCN to reason over heterogeneous graphs [25]. Xiong et al. achieved

better performance by applying graph attention networks [14]. Cao et al. proposed a bi-directional attention entity graph convolutional network [26]. These models use r-GCN [13], which does not consider the semantic relation information, or use graph attention networks to assign static weights. Different from these models, ref. [15] proposes a dynamic relation strategy, which dynamically updates relation states during the reasoning process. Documents unrelated to the complex query may affect the accuracy of the model. In the “select, answer, and explain” (SAE) model proposed by Tu et al. [27], BERT [28] acts as the encoder in the selection module. Then a sentence extractor is applied to the output of BERT to obtain the sequential output of each sentence with precalculated sentence start and end indices, to filter out answer-unrelated documents and thus reduce the amount of distraction information. The selected answer-related documents are then input to a model, which jointly predicts the answer and supporting sentences. Concurrently to the SAE model, Bhargav et al. [29] used a two-stage BERT-based architecture to first select the supporting sentence and then used the filtered supporting sentence to predict the answer. The upstream side of Jiang et al.’s [23] proposed model is the Document Explorer to iteratively address relevant documents. Han et al. [15] proposed two-phase hypergraph-based reasoning with dynamic relations which explicitly learns and updates relation information and dynamically concentrates on different relations at different hops.

2.2. Hypergraph Convolutional Networks

Feng et al. [30] proposed a hypergraph neural network, which replaces the general graph with a hypergraph structure, effectively encoding the higher-order data correlation. Bai et al. [31] further enhanced the representational learning ability by using attention modules. Yadati, N. et al. [32] proposed a new method of training a GCN on a hypergraph using tools from the spectral theory of hypergraphs and applying the method to the problems of SSL (hypergraph-based semi-supervised learning) and combinatorial optimization on real-world hypergraphs. Zhang et al. [33] developed a new self-attention-based graph neural network applicable to homogeneous and heterogeneous hypergraphs with variable hyperlink sizes. Han et al. [15] proposed a directed hypergraph convolutional network that incorporates direction information into HGNN to deal with a directed knowledge graph.

2.3. Hyperbolic Neural Networks

Hyperbolic space has always been a popular research domain in mathematics. Some works have been conducted to explore the treelike structure of graphs [34,35] and the relations between hyperbolic space and hierarchical data such as languages and complex networks [36,37]. Such works have demonstrated the consistency between real-world scale-free and hierarchical data and the hyperbolic space, providing a theoretical basis for recent works which apply hyperbolic space to various tasks including link prediction, node classification and recommendation. Some researchers apply hyperbolic space to traditional metric learning approaches such as HyperBPR [38] and HyperML [39]. Some try to adopt hyperbolic space to neural networks and define hyperbolic neural network operations, producing powerful models such as hyperbolic neural networks [40], hyperbolic graph neural networks [41] and hyperbolic convolutional neural networks [17]. Meanwhile, ref. [42] provides a scalable hyperbolic recommender system for industry use. Ref. [43] applies hyperbolic space to heterogeneous networks for link prediction tasks. Ref. [44] applies hyperbolic space to next-POI recommendation. Ref. [45] proposes a path-based recommendation approach with hyperbolic embeddings, etc.

3. Methods

3.1. Hyperbolic Directed Hypergraph Convolutional Networks

In this section, we are going to introduce the directed hypergraph convolutional network constructed on hyperbolic space. Definitions of notations used in the text are shown in the Table 1.

3.1.1. Undirected Hypergraph Convolutional Network

We first introduce undirected hypergraph convolutional neural networks. Different from simple graphs, hyperedges in a hypergraph may contain two or more vertices. A hypergraph can be defined as $\mathcal{G} = (V, E, W)$, which includes a vertex set V , a hyperedge set E and each hyperedge is assigned with a weight by W which is a diagonal matrix whose diagonal lines are the weights of each hyperedge. We use a $|V| \times |E|$ incidence matrix H to denote a hypergraph $\mathcal{G}$, and H can be concretely expressed as:

$$h(v, e) = \begin{cases} 1 & v \in e \\ 0 & v \notin e \end{cases} \quad (1)$$

For every vertex $v \in V$, the degree can be defined as $d(v) = \sum_{e \in E} \omega(e)h(v, e)$, For every hyperedge $e \in E$, the degree can be defined as $d(e) = \sum_{v \in V} \omega(v)h(v, e)$, D_e and D_v denote the diagonal matrices of the hyperedge degrees and the vertex degrees. We let $\Theta = D_v^{-\frac{1}{2}} H W D_e^{-1} H^T D_v^{\frac{1}{2}}$ and $\Delta = I - \Theta$ is defined as the hypergraph Laplacian [46], according to the Laplacian expression form, it can be known that it is a symmetric semidefinite matrix. It can be obtained by eigenvalue decomposition of symmetric positive semidefinite matrices that $\Delta = \Phi \Lambda \Phi^T$; [30] use the eigen vectors as the Fourier bases and the eigenvalues as frequencies to express the spectral convolution as:

$$g * X = \Phi((\Phi^T g) \otimes (\Phi^T X)) = \Phi g(\Delta) \Phi^T X \quad (2)$$

Ref. [30] then approximate the above equation by Chebyshev polynomials, modify inside parameters appropriately and finally formulate hyperedge convolution as:

$$X^{(l+1)} = D_v^{-1} H W D_e^{-1} H^T X^{(l)} P \quad (3)$$

where $X^{(l)}$ is the node feature of the hyperedge at layer l , and P is the learnable parameter.

3.1.2. Hyperbolic Directed Hypergraph Convolutional Neural Network

As shown in Section 3.1.1, the derivation process of hyperedge convolution is based on the eigenvalue decomposition of the Laplacian matrix of the hypergraph. In an undirected hypergraph, the Laplacian matrix is a symmetric positive semidefinite matrix because the degree matrices of its vertices and hyperedges are unique, so the eigenvalue decomposition always works. However, for directed hypergraphs, due to hyperedges having direction, the degree matrices of vertices and hyperedges should be divided into out-degree matrix and in-degree matrix, and according to the random walk explanation of spectral hypergraph partitioning [46], the specific representation of the Laplacian matrix of a directed hypergraph should be as follows:

$$\Delta = I - \Theta = I - D_v^{tail^{-1}} H^{tail} W D_e^{head^{-1}} H^{head^T} \quad (4)$$

where D_v^{tail} and D_e^{head} are the diagonal matrices of tail degrees of nodes and head degrees of hyperedges, H^{tail} and H^{head} stand for the tail and head incidence matrices.

$$H^{tail}(i, j) = \begin{cases} 1 & v_i \in e_j^{tail} \\ 0 & v_i \notin e_j^{tail} \end{cases} \quad (5)$$

$$H^{head}(i, j) = \begin{cases} 1 & v_i \in e_j^{head} \\ 0 & v_i \notin e_j^{head} \end{cases} \quad (6)$$

Since the two incidence matrices are generally different in a directed hypergraph, the Laplacian matrix is often not a symmetric matrix; as a result, the directed hypergraph

modeled on the knowledge base may not be able to derive the hyperedge convolution as the undirected hypergraph shown in Section 3.1.1, and this will produce calculation error to some extent.

Aiming at solving the problem above and the sparsity issue in hypergraphs modeled on a knowledge base, we apply the variant forms of GCN on the hyperbolic space [17] to the directed hypergraph and obtain the matrix form of the directed hypergraph convolution network on the hyperbolic space. The directed hypergraph convolution operations on hyperbolic spaces aggregate each representation vector individually in vector dimensions, without the aforementioned problem of symmetry of the Laplacian matrix of directed hypergraphs. The specific process is as follows:

We first transform the initial item features from Euclidean space to hyperbolic space H^K , and then we feed the initial hyperbolic item embeddings to learn item embeddings. For the hyperbolic space, we set $\alpha := \{\sqrt{K}, 0, 0, \dots, 0\} \in H^K$ as the north pole in H^K , and the negative curvature of the hyperboloid manifold is $-\frac{1}{K}$. Then the initial item features in hyperbolic space can be deduced from Euclidean space as follows:

$$x^{(0,H)} = \exp_{\alpha}^K((0, x^{(0,E)})) = (\sqrt{K} \cosh(\frac{\|x^{(0,E)}\|_2}{\sqrt{K}}), \sqrt{K} \sinh(\frac{\|x^{(0,E)}\|_2}{\sqrt{K}}) \frac{x^{(0,E)}}{\|x^{(0,E)}\|_2}) \quad (7)$$

where $x^{(0,H)}$ and $x^{(0,E)}$ are the initial hyperbolic embedding and the initial Euclidean embedding, respectively. For the directed hypergraph convolutional network, when updating entity representation through the convolutional layer, we first aggregate the head entities in directed hyperedges to obtain the representation of relations, and then accumulate relation representation containing the same tail entity to obtain the representation of the tail entity so as to continuously update the representation of entities. The representation can be aggregated via the following convolutional operation in hyperbolic space:

$$r_i^{L,H} = \exp_{\alpha}^K(\sum_{j \in e_{ihead}} M_{ij} \log_{\alpha}^K(x_j^{L,H})) \quad (8)$$

where $r_i^{L,H}$ is the hyperbolic hidden embedding of relation e_i in the L -th layer after aggregation, the node j 's hyperbolic embedding is transformed to Euclidean embedding via $\log_{\alpha}^K$, so the Euclidean-based sum and add operations are available. $\exp_{\alpha}^K$ aims to transform the Euclidean-based embedding to hyperbolic embedding. M_{ij} is the projecting weight defined as follows:

$$M_{ij} = \text{softmax}_{j \in e_{ihead}^H} (MLP(\log_{\alpha}^K(x_i^{L,H}) || \log_{\alpha}^K(x_j^{L,H}))) \quad (9)$$

Accordingly, we can write an expression that evaluates the tail entity representation:

$$x_t^{L+1,H} = \exp_{\alpha}^K(\sum_{s \in e_{tail}} M_{ts} \log_{\alpha}^K(r_s^{L,H})) \quad (10)$$

where $x_t^{L,H}$ is the hyperbolic embedding of tail entity x_t in the L -th layer, e_{tail} stands for the directed hyperedge containing the tail entity x_t .

Because $\exp_{\alpha}^K$ and $\log_{\alpha}^K$ are the inverse of each other, the total convolution is as follows:

$$x_t^{L+1,H} = \exp_{\alpha}^K(\sum_{s \in e_{tail}} M_{ts} w_r^L (\sum_{j \in e_{ihead}} M_{ij} \log_{\alpha}^K(x_j^{L,H}))) \quad (11)$$

The formula cancels out the adjacent inverse operation $\exp_{\alpha}^K$ and $\log_{\alpha}^K$, and applies max pooling to obtain weights for each relation,

$$w_r^L = \text{softmax}(\text{MaxPooling}(R_i^{L,H})) \quad (12)$$

To facilitate the formulation of the model, we rewrite it in matrix form:

$$X^{L+1,H} = \exp_{\alpha}^K(M_{tail}WM_{head}\log_{\alpha}^K(X^{L,H})) \quad (13)$$

where $X^{L+1,H}$ and $X^{(L,H)}$ are the entity representation matrix in the $L + 1$ -th and L -th layer, respectively. W stands for a diagonal matrix of hyperedge weights, M_{head} and M_{tail} are the aggregate matrices of the head part and tail part of directed hyperedges.

3.2. Model

In this section, we are going to introduce the concrete model for the multi-hop knowledge base QA task. The overview of how the model works is shown in Figure 2.

Table 1. Descriptions of notations used in the following parts.

Symbol	Definition
H	a $ V \times E $ incidence matrix
D_e	a diagonal matrices of the hyperedge degrees
D_v	a diagonal matrices of the vertex degrees
Δ	the hypergraph Laplacian
Φ	the eigenvectors of the hypergraph Laplacian
$\otimes$	the element-wise Hadamard product
H^K	a hyperboloid manifold which negative curvature is $-\frac{1}{K}$
α	the north pole in H^K
$\exp_{\alpha}^K$	the exponential map of the hyperboloid model
$\log_{\alpha}$	the logarithmic map of the hyperboloid model
$\cosh$	hyperbolic cosine function
$\sinh$	hyperbolic sine function
M_{head}	the aggregate matrix of the head-part of directed hyperedges
M_{tail}	the aggregate matrix of the tail-part of directed hyperedges

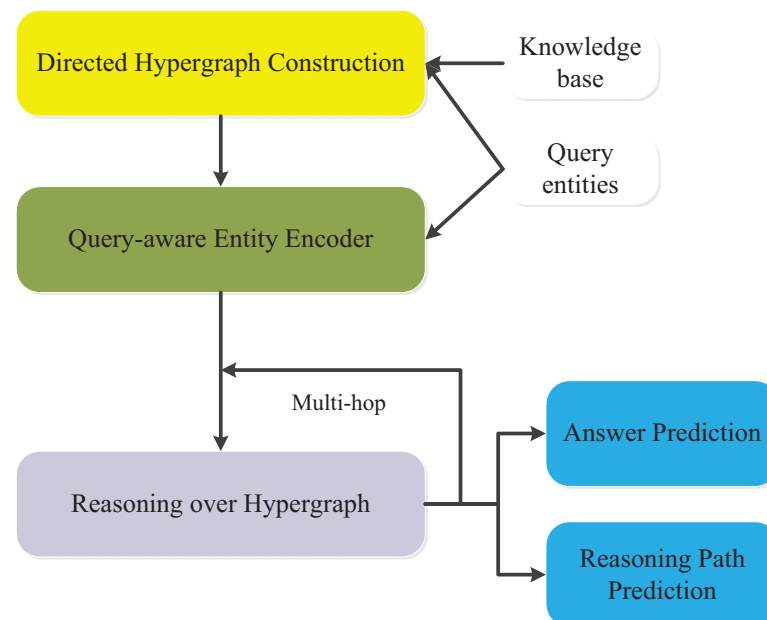


Figure 2. An overview of how the model works.

3.2.1. Query-Aware Entity Encoder

The query-aware entity encoder encodes entities and relations in questions and their potential related entities in knowledge base to vector representation. Let L_q , L_e and L_r

respectively denote the embedding matrices of question, entities and relations; we begin by encoding each question using bidirectional Gated Recurrent Units (GRUs) [47].

$$E_q = \text{BiGRU}(\tanh(W_q L_q + b_q)) \quad (14)$$

We then follow the work of [21]; we employ co-attention to learn query-aware entity representation,

$$A_{eq} = L_e(H_q^T) \quad (15)$$

$$C_e = \text{softmax}(A_{eq})E_q \quad (16)$$

$$C_q = \text{softmax}(A_{eq}^T)L_e \quad (17)$$

$$D_e = \text{BiGRU}(\text{softmax}(A_{eq})C_q) \quad (18)$$

$$E_{attn} = f_c([C_e; D_e]) \quad (19)$$

where softmax stands for column-wise normalization, f_c is a linear network which converts 2h dimension to h dimension.

3.2.2. Reasoning over Hypergraph

According to the property that relation embedding can be obtained in the intermediate processes of hyperbolic directed hypergraph convolutional networks, we separate the hyperbolic directed hypergraph convolutional networks into two steps. Specifically, the model first collects the features of nodes onto the connected hyperedges and explicitly represents the learning relationship. Then, it dynamically assigns the weight of the relationship according to the similarity between the problem and the relationship, and predicts the current relation. Finally, the node status is updated through the connection relationship information. The specific process is as follows.

Firstly, assuming that the current hop is l , we use a linear network to concatenate the node status obtained by the previous $l - 1$ layer and the input entity representation of the current hop, and then map it onto the hyperbolic space.

$$U_e^{l,H} = \exp_{\alpha}^K(f_{in}([E^{(l-1)}; E_{attn}])) \quad (20)$$

where operator $[;]$ is column-wise concatenation. Then the model learns the relation representation $R^{l,H}$ by aggregating the connected head entity feature.

$$U_r^{l,H} = M_{head} \log_{\alpha}^K(U_e^{l,H}) P_r \quad (21)$$

where P_r is the relation-specific learnable parameter, we then use a linear network to concatenate the relation representation obtained by the previous $l - 1$ layer and $U_r^{l,H}$ to obtain a representation of relations at hop l .

$$R^{l,H} = f_r([R^{(l-1),H}; U_r^{l,H}]) \quad (22)$$

After that, we apply w_r^l in (12) to obtain weights for each relation and the diagonal matrix W of edge weights is $W = \text{diag}(w_r^l)$. The weight of the dynamic allocation relation depends on the updated relation representation hop-by-hop. This model predicts the current relation based on the relation weight.

Finally, the model adaptively updates entity states by accumulating connected relation feature

$$E^l = \exp_{\alpha}^K(M_{tail} W R^{l,H} P_e) \quad (23)$$

where P_e is the entity-specific learnable parameter.

3.3. Training

For an L hop question, we sum the entity representation of each layer to obtain the final representation and use a linear layer f_{ans} to predict the answer distribution.

$$P = \sigma(f_{ans}(\sum_{l=1}^L E^{(l)})) \quad (24)$$

where σ is the sigmoid function.

Since the model needs to predict both the answer to the question and the reasoning path, the loss function consists of two parts, one is binary cross-entropy loss of the final answer prediction, the other is the negative log likelihood of the intermediate prediction of the reasoning path. The specific expression of the loss function is as follows:

$$L = - \sum_{i=1}^n (y_i \log(p_i) + (1 - y_i) \log(1 - p_i)) + \lambda \times \sum_{l=1}^{L+1} (-\log(w_r^l(r_l^*))) \quad (25)$$

where y_i is the golden distribution over entities. r_l^* is the golden relation index at hop L . λ is a hyper parameter to balance the two terms.

4. Experiment

This section reports the experiments.

4.1. Experiment Setup

We detail the adopted datasets, evaluation metrics, parameters and baselines.

4.1.1. Datasets

We use two single answer KBQA datasets and two large-scale multi-answer KBQA datasets for the multi-hop KBQA task. We briefly outline these datasets in Table 2.

Table 2. Statistics of the hypergraph datasets used in the experiments.

Datasets	PQL-2H	PQL-3H	MetaQA-1H	MetaQA-2H
Number of Questions	1594	1031	116,049	148,724
Number of Entities in knowledge base	5034	5034	40,128	40,128
Number of Relations in knowledge base	364	364	9	9

- **PQL-2H** [48]: PQL-2H is a single answer KBQA dataset, which includes a knowledge base containing 5035 entities and 364 relationships, and a two-hop question set containing 1594 two-hop questions. These questions can be answered by following the reasoning path consisting of several relations and intermediate entities. The path has been given.
- **PQL-3H** [48]: PQL-3H is a single-answer KBQA dataset, which includes the same knowledge base with 5035 entities and 364 relations as PQL-2H, and a three-hop question set with 1031 three-hop questions. The characteristics of questions and the reasoning path are the same as PQL-2H.
- **MetaQA-1H** [49]: MetaQA-1H contains 116,045 questions for single-hop reasoning QA and the knowledge base in the dataset contains 40,128 entities and nine relations. To test QA systems in more realistic (and more difficult) scenarios, MetaQA-1H also

provides neural-translation-model-paraphrased datasets, and text-to-speech-based audio datasets.

- **MetaQA-2H** [49]: MetaQA-2H contains 148,724 questions for two-hop reasoning and the knowledge base in the dataset contains 40,128 entities and nine relations. MetaQA-2H provides neural-translation-model-paraphrased datasets, and text-to-speech-based audio datasets just like MetaQA-1H.

4.1.2. Metrics and Parameters

We test the effectiveness of the model in four datasets. The total questions in datasets are divided into three parts: 70% for training, 10% for validation and 20% for testing. We evaluate the experiment results via two standard metrics: F1 and Hits@1. The F1 value is an overall evaluation of the precision and recall, which can evaluate the performance of the model well, while Hits@1 measures the proportion of top 1 rankings. The aim of the training is to achieve high F1 and Hits@1.

The reported results are given for the best set of hyper-parameters evaluated on the validation set for each model after grid search on the following values: embedding size $\in \{100, 200, 300, 400, 500\}$, learning rate $\in \{1, 0.1, 0.01, 0.001\}$, λ and dropout are set to 1 and 0.4.

4.1.3. Baselines

We compare HDH-GCN with the following baselines:

- **KVMemNet** [50]: This is an end-to-end memory network which divides the memory into two parts, the key memory stores the head entity and relation, and the value memory stores the tail entity.
- **IRN** [48]: This is an interpretable reasoning network, which uses a hop-by-hop reasoning process and answers questions based on knowledge maps.
- **VRN** [49]: An end-to-end variational learning algorithm is proposed, which can effectively solve the multi hop reasoning problem and simultaneously deal with the noise in the problem
- **GraftNet** [12]: Text information and entities are introduced to construct a graph, and GCN is applied to reasoning.
- **SGReader** [14]: This also combines the unstructured text and knowledge graph to figure out the incompleteness of the knowledge graph. The model employs graph attention to reason effectively.
- **2HR-DR** [15]: This models the entities extracted from questions and their related relationships and entities in the knowledge base into directed hypergraphs, then uses Directed Hypergraph Convolutional Networks to predict relations hop-by-hop and form a sequential relation path to make the reasoning interpretable.

4.2. Results of Main Experiment

Tables 3 and 4 show the main experiment results for two kinds of multi-hop KBQA datasets. The highest scores are in bold. As shown in Table 3, we can find out that our proposed HDH-GCN can achieve optimal results under Hit@1 measurement standards (there is only one answer for each question in PQL datasets, so we only adopt Hits@1 for evaluation). For the rest of the datasets, except for the F1 value of HDH-GCN on the MetaQA-1H dataset, which does not exceed the baseline model, other evaluation indexes have been improved to some extent, as shown in Table 4. Specifically, HDH-GCN achieves an improvement for PQL-2H which is 0.9% higher than the second best model. It also obtains good result on PQL-3H, 1.2% higher than the second best one. Table 4 demonstrates the performance of the baseline methods and HDH-GCN on the MetaQA-1H dataset; our model improves Hits@1 by 1.8% and obtains competitive F1. For MetaQA2-Hop, we improve Hits@1 and F1 by 0.3% and 0.8%, respectively.

Table 3. Results of Hit@1 on PQL-2H and PQL-3H.

Model	PQL-2H	PQL-3H
KVMemNet	0.690	0.617
IRN	0.725	0.710
GraftNet	0.707	0.910
SGReader	0.719	0.893
2HR-DR	0.755	0.921
HDH-GCN	0.764	0.933

Table 4. Results of Hit@1 and F1 on MetaQA-1H and MetaQA-2H.

Model	MetaQA-1H		MetaQA-2H	
	Hit@1	F1	Hit@1	F1
KVMemNet	0.958	-	0.251	-
VRN	0.975	-	0.898	-
GraftNet	0.970	0.910	0.948	0.727
SGReader	0.967	0.960	0.807	0.798
2HR-DR	0.988	0.973	0.937	0.814
HDH-GCN	0.990	0.968	0.951	0.822

First of all, compared with models based on a knowledge base modeled on a simple graph, our model reconstructs the knowledge graph to hypergraph structure which fully considers the high-order data correlation. Meanwhile, we dynamically concentrate on relation information at different hops by performing loop operations for each hop of inference to guide the model to follow the golden relation path and select the final answers, so we can introduce the information of the intermediate reasoning path into our model to supervise the model focus on the dynamic relations at different hops.

When comparing with the directed hypergraph-based model 2HR-DR, the improvement of both evaluation values on the two PQL datasets is more obvious than on the other datasets. The reasons why our method performs better include: (1) The directed hypergraph is modeled on hyperbolic space, which effectively reduces the sparsity of the incidence matrix of the directed hypergraph; this will reduce the scale of matrix calculation during training and reduce the inadequacy of training. This also explains why the results on the two PQL datasets are better than on MetaQA-1H and MetaQA-2H, the number of relationships in PQL is much higher than in MetaQA, which leads to more obvious matrix sparsity problems (the number of relations in MetaQA is small, and each relation can relate to many entities). (2) In the convolutional network of a directed hypergraph, there is a computational process of eigenvalue decomposition of the asymmetric Laplacian matrix, but the asymmetric matrix may not be able to carry out the eigenvalue decomposition, which leads to the forced eigenvalue decomposition of the matrix that cannot be diagonalized similarly in the training process, and the training error is always caused. The problem can be solved effectively by deforming the Laplace matrix in hyperbolic space. This problem is evident in the single-answer QA task, while the multi-answer QA task will dilute the influence of this problem to a certain extent when calculating the F1 value, which also explains why the F1 value of HDH-GCN is not improved in MetaQA-1H.

4.3. Parameter Analysis

Embedding size is a significant factor in KBQA models, determining the performance of the model to a large extent. Hence, we will analyze the results obtained by the model on PQL-2H in different embedding sizes to investigate its impact. First, according to the Figure 3a, HDH-GCN outperforms other methods when the dimension is {100, 200, 300, 400}. The Hit@1 of HDH-GCN increases sharply with the early stage of increasing the embedding size and becomes smooth after the embedding size increases to 400. The Hit@1 of 2HR-DR is almost identical to TF-DHP's from the start; however, because the sparsity

issue becomes intense as the dimension increases, it cannot remain smooth like HDH-GCN when the embedding size increases. After the embedding size increases to a certain extent, 2HR-DR's Hit@1 will decrease slightly. For other methods, since the knowledge base is not modeled on the hypergraph, the sparsity issue has no obvious effect on the training results of higher dimensions; however, due to the reasons mentioned above, its results in each dimension are not as superior as HDH-GCN. We also count the Hit@1 results on PQL-2H of each training session. In the Figure 3b, we compare the Hit@1 between HDH-GCN and 2HR-DR on model training. 2HR-DR is stopped early around 35 epochs because of not updating Hit@1 for 10 epochs, so the line is not complete. HDH-GCN always achieves better performance, and is still updating until around 34 epochs.

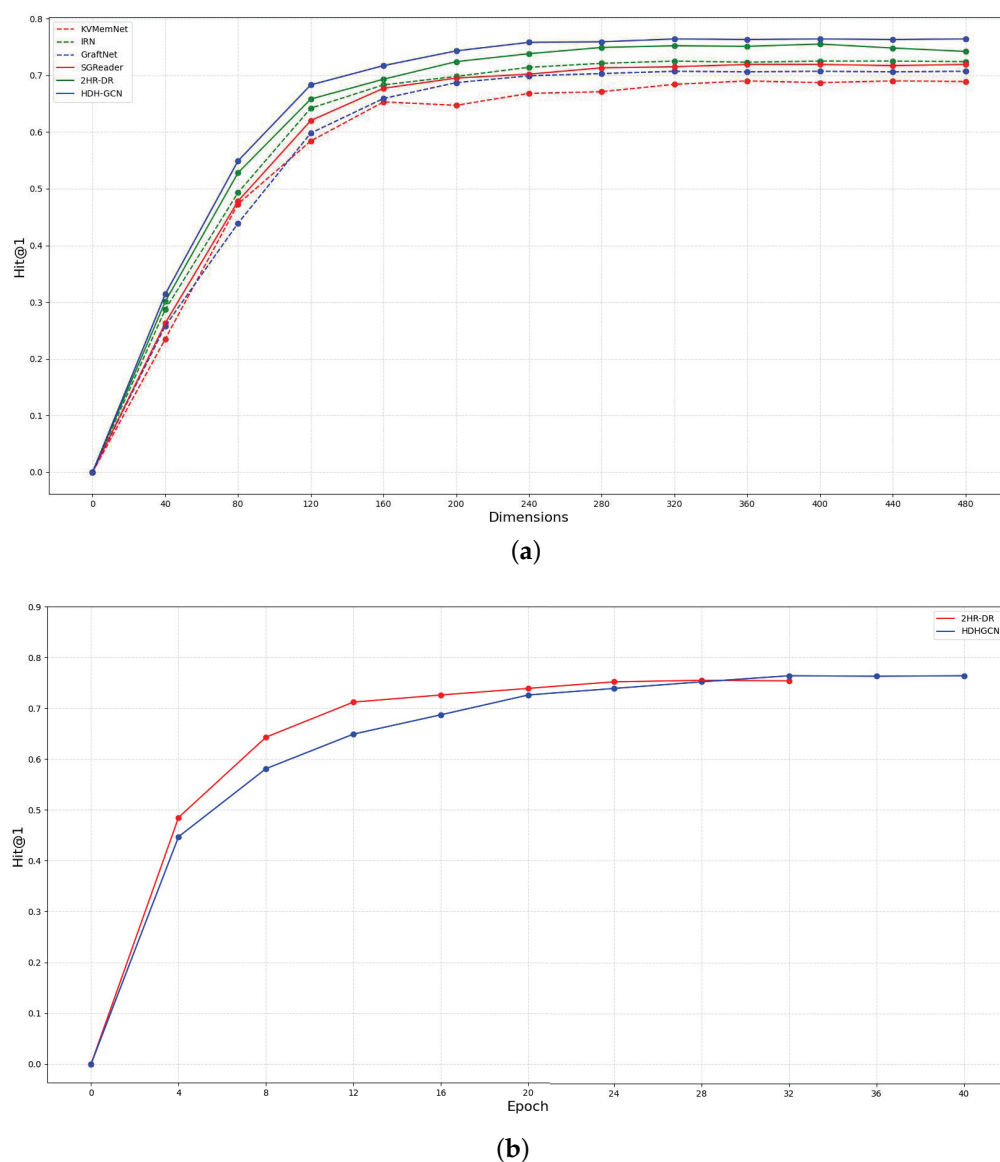


Figure 3. (a) Hit@1 over different embedding sizes of knowledge base QA models, evaluated on PQL-2H. (b) Hit@1 over different training epochs, evaluated on PQL-2H.

4.4. Approximate Training Time Comparison

On the two kinds of datasets PQL and MetaQA, HDH-GCN takes around 75 min and 3 h, respectively, of training time, while 2HR-DR takes around 2 h and 3 h, respectively. All were run on a GeForce GTX 1080 super GPU machine with Python 3.

4.5. Case Study

As Figure 4 shows, we give an exemplar question from PQL-2H and its corresponding reasoning path and triples in KB. It is clear that HDH-GCN has the ability to predict relations hop-by-hop and stop reasoning automatically. For question “Who is the singer of the theme song of the movie “Titanic”?”, the model firstly detects the relation “theme song”, then “singer” successively and finally meets <STOP> to end the reasoning process. From the “path” in Figure 4, we can observe our model’s predicted relation path (Theme song $\rightarrow$ Singer $\rightarrow$ <STOP>).

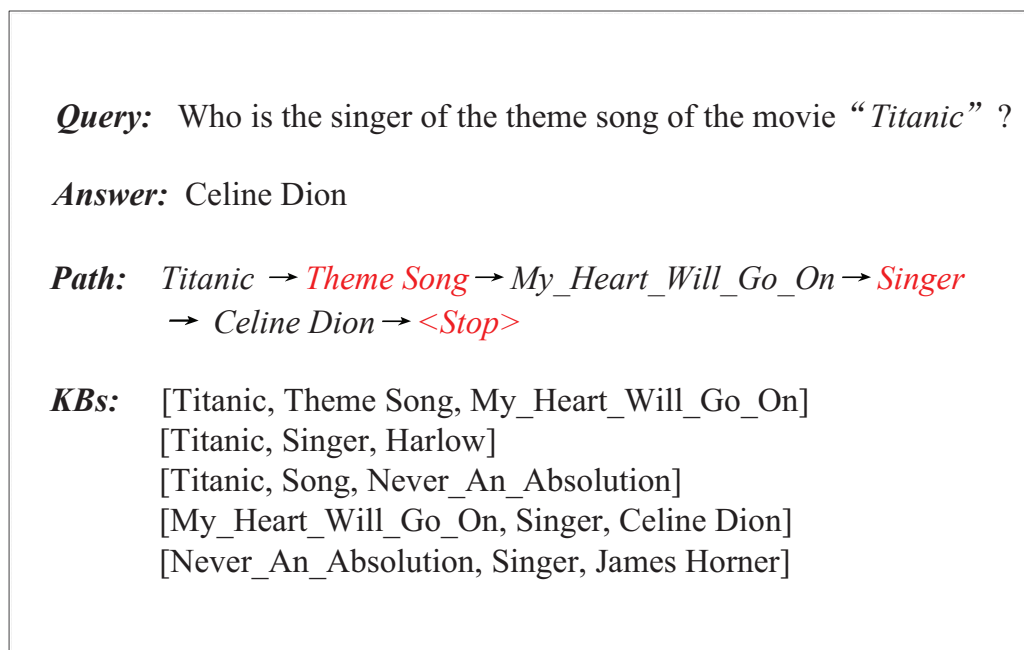


Figure 4. An exemplar of a two-hop query in a KBQA dataset; the figure shows the reasoning path and triples related to entities in the query, and graphically shows how the model reaches the answer to the question through the reasoning path.

5. Conclusions and Future Work

In this paper, we introduce HDH-GCN, a novel model for multi-hop KBQA tasks. We model the directed hypergraph convolutional network in hyperbolic space, which effectively reduces the influence of the sparsity issue on the model effect. Our model can improve the accuracy of the multi-hop knowledge base question-answering task, and has application value in text question answering, human–computer interactions and other fields. The experimental results verify the advantages of HDH-GCN in both single-answer questions and multi-answer question datasets.

In the future, we will study the multi-hop knowledge base question-answering task in multi-modal data, study the possibility of modeling a multi-modal knowledge base by directed hypergraph and explore the possible application prospect.

Author Contributions: Conceptualization, G.X. and J.L.; methodology, G.X.; software, G.X. and J.L.; validation, G.X., J.L. and Z.T.; formal analysis, G.X.; investigation, G.X., J.L. and Y.Y.; data curation, G.X.; writing—original draft preparation, G.X.; writing—review and editing, G.X., J.L., Z.T. and B.G.; visualization, G.X. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by NSFC under grants Nos.61902417 and 71971212.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Huang, X.; Zhang, J.; Li, D.; Li, P. Knowledge Graph Embedding Based Question Answering. In Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, WSDM 2019, Melbourne, VIC, Australia, 11–15 February 2019; Culpepper, J.S., Moffat, A., Bennett, P.N., Lerman, K., Eds.; ACM: New York, NY, USA, 2019; pp. 105–113.
- Rajpurkar, P.; Zhang, J.; Lopyrev, K.; Liang, P. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, TX, USA, 1–4 November 2016; Su, J., Carreras, X., Duh, K., Eds.; The Association for Computational Linguistics: Stroudsburg, PA, USA, 2016; pp. 2383–2392.
- Rajpurkar, P.; Jia, R.; Liang, P. Know What You Don’t Know: Unanswerable Questions for SQuAD. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, 15–20 July 2018; Gurevych, I., Miyao, Y., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; Volume 2: Short Papers, pp. 784–789.
- Kociský, T.; Schwarz, J.; Blunsom, P.; Dyer, C.; Hermann, K.M.; Melis, G.; Grefenstette, E. The NarrativeQA Reading Comprehension Challenge. *Trans. Assoc. Comput. Linguist.* **2018**, *6*, 317–328. [\[CrossRef\]](#)
- Reddy, S.; Chen, D.; Manning, C.D. CoQA: A Conversational Question Answering Challenge. *Trans. Assoc. Comput. Linguist.* **2019**, *7*, 249–266. [\[CrossRef\]](#)
- Cao, X.; Liu, Y. Coarse-grained decomposition and fine-grained interaction for multi-hop question answering. *J. Intell. Inf. Syst.* **2022**, *58*, 21–41. [\[CrossRef\]](#)
- Xu, K.; Reddy, S.; Feng, Y.; Huang, S.; Zhao, D. Question Answering on Freebase via Relation Extraction and Textual Evidence. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, Berlin, Germany, 7–12 August 2016; The Association for Computer Linguistics: Stroudsburg, PA, USA, 2016; Volume 1: Long Papers.
- Lin, X.V.; Socher, R.; Xiong, C. Multi-Hop Knowledge Graph Reasoning with Reward Shaping. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; Riloff, E., Chiang, D., Hockenmaier, J., Tsujii, J., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 3243–3253.
- Zhou, M.; Huang, M.; Zhu, X. An Interpretable Reasoning Network for Multi-Relation Question Answering. In Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, NM, USA, 20–26 August 2018; Bender, E.M., Derczynski, L., Isabelle, P., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 2010–2022.
- Xu, K.; Lai, Y.; Feng, Y.; Wang, Z. Enhancing Key-Value Memory Neural Networks for Knowledge Based Question Answering. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, 2–7 June 2019; Burstein, J., Doran, C., Solorio, T., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; Volume 1 (Long and Short Papers), pp. 2937–2947.
- Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; Yu, P.S. A Comprehensive Survey on Graph Neural Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 4–24. [\[CrossRef\]](#) [\[PubMed\]](#)
- Sun, H.; Dhingra, B.; Zaheer, M.; Mazaitis, K.; Salakhutdinov, R.; Cohen, W.W. Open Domain Question Answering Using Early Fusion of Knowledge Bases and Text. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; Riloff, E., Chiang, D., Hockenmaier, J., Tsujii, J., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 4231–4242.
- Schlichtkrull, M.S.; Kipf, T.N.; Bloem, P.; van den Berg, R.; Titov, I.; Welling, M. Modeling Relational Data with Graph Convolutional Networks. In Proceedings of the Semantic Web—15th International Conference, ESWC 2018, Heraklion, Greece, 3–7 June 2018; Gangemi, A., Navigli, R., Vidal, M., Hitzler, P., Troncy, R., Hollink, L., Tordai, A., Alam, M., Eds.; Lecture Notes in Computer Science; Springer: Berlin, Germany, 2018; Volume 10843, pp. 593–607.
- Xiong, W.; Yu, M.; Chang, S.; Guo, X.; Wang, W.Y. Improving Question Answering over Incomplete KBs with Knowledge-Aware Reader. In Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, 28 July–2 August 2019; Korhonen, A., Traum, D.R., Màrquez, L., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; Volume 1: Long Papers, pp. 4258–4264.
- Han, J.; Cheng, B.; Wang, X. Two-Phase Hypergraph Based Reasoning with Dynamic Relations for Multi-Hop KBQA. In Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020, Yokohama, Japan, 11–17 July 2020; Bessiere, C., Ed.; pp. 3615–3621.
- Zhang, S.; Chen, H.; Ming, X.; Cui, L.; Yin, H.; Xu, G. Where are we in embedding spaces? A Comprehensive Analysis on Network Embedding Approaches for Recommender Systems. *arXiv* **2021**, arXiv:2105.08908.
- Chami, I.; Ying, Z.; Ré, C.; Leskovec, J. Hyperbolic Graph Convolutional Neural Networks. In Proceedings of the Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, Vancouver, BC, Canada, 8–14 December 2019; Wallach, H.M., Larochelle, H., Beygelzimer, A., d’Alché-Buc, F., Fox, E.B., Garnett, R., Eds.; 2019; pp. 4869–4880.
- Bordes, A.; Weston, J.; Usunier, N. Open Question Answering with Weakly Supervised Embedding Models. In Proceedings of the Machine Learning and Knowledge Discovery in Databases—European Conference, ECML PKDD 2014, Nancy, France, 15–19 September 2014; Part I; Lecture Notes in Computer Science; Calders, T., Esposito, F., Hüllermeier, E., Meo, R., Eds.; Springer: Berlin, Germany, 2014; Volume 8724, pp. 165–180.

19. Hao, Y.; Zhang, Y.; Liu, K.; He, S.; Liu, Z.; Wu, H.; Zhao, J. An End-to-End Model for Question Answering over Knowledge Base with Cross-Attention Combining Global Knowledge. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, BC, Canada; 30 July–4 August 2017; Barzilay, R., Kan, M., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2017; Volume 1: Long Papers, pp. 221–231.
20. Min, S.; Zhong, V.; Socher, R.; Xiong, C. Efficient and Robust Question Answering from Minimal Context over Documents. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, 15–20 July 2018; Gurevych, I., Miyao, Y., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; Volume 1: Long Papers, pp. 1725–1735.
21. Zhong, V.; Xiong, C.; Keskar, N.S.; Socher, R. Coarse-grain Fine-grain Coattention Network for Multi-evidence Question Answering. In Proceedings of the 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, 6–9 May 2019.
22. Kundu, S.; Khot, T.; Sabharwal, A.; Clark, P. Exploiting Explicit Paths for Multi-hop Reading Comprehension. In Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, 28 July–2 August 2019; Korhonen, A., Traum, D.R., Màrquez, L., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; Volume 1: Long Papers, pp. 2737–2747.
23. Jiang, Y.; Joshi, N.; Chen, Y.; Bansal, M. Explore, Propose, and Assemble: An Interpretable Model for Multi-Hop Reading Comprehension. In Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, 28 July–2 August 2019; Korhonen, A., Traum, D.R., Màrquez, L., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; Volume 1: Long Papers, pp. 2714–2725.
24. Sun, H.; Bedrax-Weiss, T.; Cohen, W.W. PullNet: Open Domain Question Answering with Iterative Retrieval on Knowledge Bases and Text. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, 3–7 November 2019; Inui, K., Jiang, J., Ng, V., Wan, X., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; pp. 2380–2390.
25. Tu, M.; Wang, G.; Huang, J.; Tang, Y.; He, X.; Zhou, B. Multi-hop Reading Comprehension across Multiple Documents by Reasoning over Heterogeneous Graphs. In Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, 28 July–2 August 2019; Korhonen, A., Traum, D.R., Màrquez, L., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; Volume 1: Long Papers, pp. 2704–2713.
26. Cao, Y.; Fang, M.; Tao, D. BAG: Bi-directional Attention Entity Graph Convolutional Network for Multi-hop Reasoning Question Answering. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, 2–7 June 2019; Burstein, J., Doran, C., Solorio, T., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; Volume 1 (Long and Short Papers), pp. 357–362.
27. Tu, M.; Huang, K.; Wang, G.; Huang, J.; He, X.; Zhou, B. Select, Answer and Explain: Interpretable Multi-Hop Reading Comprehension over Multiple Documents. In Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, 7–12 February 2020; AAAI Press: Menlo Park, CA, USA, 2020; pp. 9073–9080.
28. Devlin, J.; Chang, M.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, 2–7 June 2019; Burstein, J., Doran, C., Solorio, T., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; Volume 1 (Long and Short Papers), pp. 4171–4186.
29. Bhargav, G.P.S.; Glass, M.R.; Garg, D.; Shevade, S.K.; Dana, S.; Khandelwal, D.; Subramaniam, L.V.; Gliozzo, A. Translucent Answer Predictions in Multi-Hop Reading Comprehension. In Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, 7–12 February 2020; AAAI Press: Menlo Park, CA, USA, 2020; pp. 7700–7707.
30. Feng, Y.; You, H.; Zhang, Z.; Ji, R.; Gao, Y. Hypergraph Neural Networks. In Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, HI, USA, 27 January–1 February 2019; AAAI Press: Menlo Park, CA, USA, 2019; pp. 3558–3565.
31. Bai, S.; Zhang, F.; Torr, P.H.S. Hypergraph convolution and hypergraph attention. *Pattern Recognit.* **2021**, *110*, 107637. [[CrossRef](#)]
32. Yadati, N.; Nimishakavi, M.; Yadav, P.; Nitin, V.; Louis, A.; Talukdar, P.P. HyperGCN: A New Method For Training Graph Convolutional Networks on Hypergraphs. In Proceedings of the Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, Vancouver, BC, Canada, 8–14 December 2019; Wallach, H.M., Larochelle, H., Beygelzimer, A., d’Alché-Buc, F., Fox, E.B., Garnett, R., Eds.; 2019; pp. 1509–1520.
33. Zhang, R.; Zou, Y.; Ma, J. Hyper-SAGNN: A self-attention based graph neural network for hypergraphs. In Proceedings of the 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, 26–30 April 2020.

34. Adcock, A.B.; Sullivan, B.D.; Mahoney, M.W. Tree-Like Structure in Large Social and Information Networks. In Proceedings of the 2013 IEEE 13th International Conference on Data Mining, Dallas, TX, USA, 7–10 December 2013; Xiong, H., Karypis, G., Thuraisingham, B., Cook, D.J., Wu, X., Eds.; IEEE Computer Society: Washington, DC, USA, 2013; pp. 1–10.
35. Chen, W.; Fang, W.; Hu, G.; Mahoney, M.W. On the Hyperbolicity of Small-World and Treelike Random Graphs. *Internet Math.* **2013**, *9*, 434–491. [[CrossRef](#)]
36. Krioukov, D.V.; Papadopoulos, F.; Kitsak, M.; Vahdat, A.; Boguñá, M. Hyperbolic Geometry of Complex Networks. *arXiv* **2010**, arXiv:1006.5169.
37. Nickel, M.; Kiela, D. Poincaré Embeddings for Learning Hierarchical Representations. In Proceedings of the Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, Long Beach, CA, USA, 4–9 December 2017; Guyon, I., von Luxburg, U., Bengio, S., Wallach, H.M., Fergus, R., Vishwanathan, S.V.N., Garnett, R., Eds.; 2017; pp. 6338–6347.
38. Vinh, T.D.Q.; Tay, Y.; Zhang, S.; Cong, G.; Li, X. Hyperbolic Recommender Systems. *arXiv* **2018**, arXiv:1809.01703.
39. Tran, L.V.; Tay, Y.; Zhang, S.; Cong, G.; Li, X. HyperML: A Boosting Metric Learning Approach in Hyperbolic Space for Recommender Systems. In Proceedings of the WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, 3–7 February 2020; Caverlee, J., Hu, X.B., Lalmas, M., Wang, W., Eds.; ACM: New York, NY, USA, 2020; pp. 609–617.
40. Ganea, O.; Bécigneul, G.; Hofmann, T. Hyperbolic Neural Networks. In Proceedings of the Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, Montréal, QC, Canada, 3–8 December 2018; Bengio, S., Wallach, H.M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R., Eds.; 2018; pp. 5350–5360.
41. Liu, Q.; Nickel, M.; Kiela, D. Hyperbolic Graph Neural Networks. In Proceedings of the Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, Vancouver, BC, Canada, 8–14 December 2019; Wallach, H.M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E.B., Garnett, R., Eds.; 2019; pp. 8228–8239.
42. Chamberlain, B.P.; Hardwick, S.R.; Wardrope, D.R.; Dzogang, F.; Daolio, F.; Vargas, S. Scalable Hyperbolic Recommender Systems. *arXiv* **2019**, arXiv:1902.08648.
43. Wang, X.; Zhang, Y.; Shi, C. Hyperbolic Heterogeneous Information Network Embedding. In Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, HI, USA, 27 January–1 February 2019; AAAI Press: Menlo Park, CA, USA, 2019; pp. 5337–5344.
44. Feng, S.; Tran, L.V.; Cong, G.; Chen, L.; Li, J.; Li, F. HME: A Hyperbolic Metric Embedding Approach for Next-POI Recommendation. In Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, 25–30 July 2020; Huang, J.X., Chang, Y., Cheng, X., Kamps, J., Murdock, V., Wen, J., Liu, Y., Eds.; ACM: New York, NY, USA, 2020; pp. 1429–1438.
45. Papadis, N.; Stai, E.; Karyotis, V. A path-based recommendations approach for online systems via hyperbolic network embedding. In Proceedings of the 2017 IEEE Symposium on Computers and Communications, ISCC 2017, Heraklion, Greece, 3–6 July 2017; IEEE Computer Society: Washington, DC, USA, 2017; pp. 973–980.
46. Zhou, D.; Huang, J.; Schölkopf, B. Learning with Hypergraphs: Clustering, Classification, and Embedding. In Proceedings of the Advances in Neural Information Processing Systems 19, Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems, Vancouver, BC, Canada, 4–7 December 2006; Schölkopf, B., Platt, J.C., Hofmann, T., Eds.; MIT Press: Cambridge, MA, USA, 2006; pp. 1601–1608.
47. Cho, K.; van Merriënboer, B.; Gülçehre, Ç.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, Doha, Qatar, 25–29 October 2014; A meeting of SIGDAT, a Special Interest Group of the ACL, Moschitti, A., Pang, B., Daelemans, W., Eds.; ACL: 2014; pp. 1724–1734.
48. Zhou, M.; Huang, M.; Zhu, X. An Interpretable Reasoning Network for Multi-Relation Question Answering. In Proceedings of the 27th International Conference on Computational Linguistics (COLING), Santa Fe, NM, USA, 20–26 August 2018; pp. 2010–2022.
49. Zhang, Y.; Dai, H.; Kozareva, Z.; Smola, A.J.; Song, L. Variational Reasoning for Question Answering With Knowledge Graph. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th Innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, LA, USA, 2–7 February 2018; McIlraith, S.A., Weinberger, K.Q., Eds.; AAAI Press: Menlo Park, CA, USA, 2018; pp. 6069–6076.
50. Miller, A.H.; Fisch, A.; Dodge, J.; Karimi, A.; Bordes, A.; Weston, J. Key-Value Memory Networks for Directly Reading Documents. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, TX, USA, 1–4 November 2016; Su, J., Carreras, X., Duh, K., Eds.; The Association for Computational Linguistics: Stroudsburg, PA, US, 2016; pp. 1400–1409.