Neurocomputing 576 (2024) 127250

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Tri-relational multi-faceted graph neural networks for automatic question tagging

Nuojia Xu [a,b], Jun Hu [c], Quan Fang [d,*], Dizhan Xue [a,b], Yongxi Li [a,b], Shengsheng Qian [a,b]

[a] State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China
[b] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, 100049, China
[c] School of Computing, National University of Singapore, 119077, Singapore
[d] School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing, 100876, China

ARTICLE INFO

Communicated by M. Wu

Keywords:
Graph Neural Networks
Community Question Answering
Question Tagging

ABSTRACT

Automatic question tagging is a crucial task in Community Question Answering (CQA) systems such as Zhihu or Quora, as it can significantly enhance the user experience by improving the efficiency of question answering and expert recommendations. Graph-based collaborative filtering models show promising performance on this task, as they can exploit not only the semantics of text content but also the existing relations between questions and tags. However, existing approaches typically encode each question into a single vector, which may not be able to capture the diverse semantic facets of questions in CQA systems. To address this challenge, we propose a novel question-tagging framework, named Tri-Relational Multi-Faceted Graph Neural Networks (TRMFG) for Automatic Question Tagging. In TRMFG, a tri-relational graph structure is designed to better model the question-tag relations. We also propose tri-relational question-tag GNN to extract hidden latent representations of questions and tags. Specially, the Multi-Faceted Question GNN helps capture the diverse semantics of questions from relevant tags. Then we build a multiple matching component to capture more complex matching patterns of the questions based on the diverse semantics. Our experimental results on three benchmark datasets demonstrate that TRMFG significantly improves question tagging performance for CQA, outperforming the state-of-the-art methods.

1. Introduction

In recent years, **Community Question Answering (CQA)** websites such as Zhihu[1] and Stack Overflow[2] have gained substantial popularity as reliable sources of information for users seeking answers to a wide variety of questions [1–4]. In CQA systems, questions are typically associated with multiple tags, which help users find the content they need. This not only aids users in discovering relevant information, but also enhances the functionality of various applications such as recommendation systems [5–7], expert-finding systems [8–10], and search engines [11–13] within CQA websites. However, a frequently encountered issue in CQA websites is the incomplete question tags assigned by users, resulting in a large number of questions lacking sufficient tags. While most CQA websites provide users with the option to tag their questions through the provided interface, it is insufficient in addressing the tagging inadequacy. For newly raised questions, only a few users may come across and tag them. What is more, when users tag the questions, they may be careless and miss some crucial tags.

To address the problem mentioned before, **Automatic Question Tagging (AQT)** has emerged as a promising solution to enhance the tags provided by users [14,15]. The definition of AQT is summarized in Fig. 1. With inputting a question with a small amount of user-assigned tags, AQT requires the question tagging model to retrieve additional related tags from the tag database.

Within CQA websites, relation information helps us learn hidden features of questions and tags. Tag-tag relations represent tag hierarchies, while question-tag relations reveal the user-assigned tags related to questions. These relations facilitate the representation of CQA website data in the form of graphs. Consequently, **GNN-based approaches** are employed for automatic question tagging. Researchers work on popular GNN-based collaborative filtering approaches to address the problem. Tian et al. [16] came up with a novel GNN-based collaborative filtering model to denoise unreliable interactions in social network graphs. Some heterogeneous graph models incorporate attention mechanisms with heterogeneous graph neural networks to learn the weights

* Corresponding author.
E-mail address: qfang@bupt.edu.cn (Q. Fang).

1 https://www.zhihu.com/
2 https://stackoverflow.com/

https://doi.org/10.1016/j.neucom.2024.127250
Received 17 August 2023; Received in revised form 31 December 2023; Accepted 7 January 2024
Available online 11 January 2024

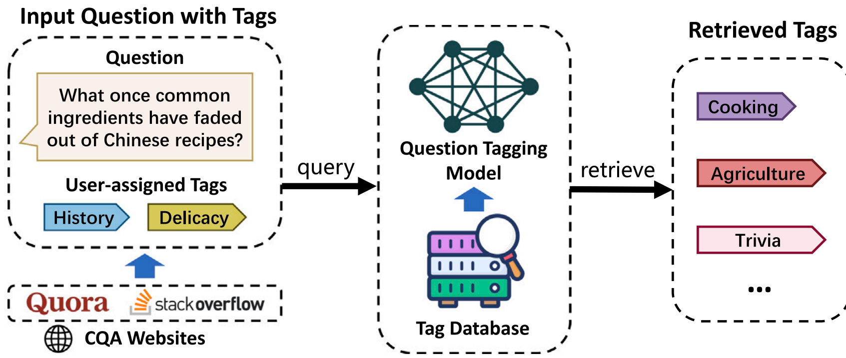
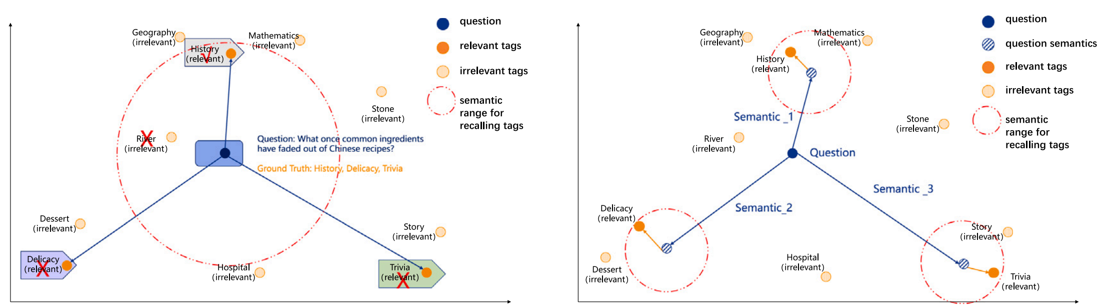


Fig. 1. The definition of AQT. Questions in CQA websites may be assigned with incomplete tags by users. So the question tagging model will help retrieve more tags for the question.



(a) Existing method.

(b) Our proposed method.

Fig. 2. Examples of questions and tags in CQA. (a) shows that existing methods focusing on only one facet of questions may fail in acquiring semantics from relevant tags for questions. (b) shows how our proposed method tempts to get multiple semantics of relevant tags by adapting multiple facets in the model.

of various relations. The heterogeneous graph transformer (HGT) [17] is a typical example. The heterogeneous graph model can learn new features for each node by utilizing various meta paths. Consequently, the HGT model can assist nodes in acquiring various types of features when utilized for recommendation tasks. Nie et al. [18] attempted to apply tag hierarchy to question tagging tasks, but disregarded the existing question-tag relations.

Despite numerous attempts to employ GNN-based collaborative filtering methods for question tagging tasks, the existing models still exhibit several limitations. In CQA websites, tag hierarchies are supposed to indicate both parent and child tag nodes for each tag. Diverse questions may already encompass specific concepts from child tags and require shared concepts from parent tags, and vice versa. As a result, making effective use of tag hierarchies plays an important role in the question tagging task. Additionally, questions posted on CQA websites may lack crucial semantics present in user-assigned tags, which is essential for the question tagging process. Existing methods [18,19] mainly concentrate on relations from parent tag nodes to child tag nodes, and only rewarding positive question-tag pairs instead of fully utilizing them. Inadequate utilization of these relations can result in an incomplete structure of questions and tags within CQA websites. Therefore, we have to face **Challenge 1**: how to appropriately make use of existing relations in CQA websites for further question tagging?

Moreover, traditional question tagging methods [18,19] often concentrate on the original question semantics. For instance, the question 'What once common ingredients have faded out of Chinese recipes?' has semantics of 'recipe' and 'ingredient'. So the traditional question tagging models may retrieve tags such as 'recipe' or 'ingredient' for the question. While in Fig. 2(a), its relevant tags are 'Delicacy', 'History', and 'Trivia' in CQA websites. This problem indicates that we should also take the semantics of relevant tags into consideration when retrieving more tags. Meanwhile, in common GNN-based question tagging models, as illustrated in Fig. 2(a), the question 'What once common ingredients have faded out of Chinese recipes?' may retrieve wrong tags 'River' and ignore relevant tags 'Delicacy' and 'History'. Although the question has a large range, it still fails to capture proper semantics from relevant tags. These wrong semantics will disturb the question tagging task. Therefore, we need to address **Challenge 2**: how to better learn the diverse semantics inherent in questions by relevant tags?

To address the above challenges, we propose a novel question-tagging framework named Tri-Relational Multi-Faceted Graph Neural Networks (TRMFG) for Automatic Question Tagging. To tackle **Challenge 1**, a Tri-Relational Question-Tag Graph is constructed to effectively represent and facilitate information propagation between question and tag nodes by emphasizing three crucial relations. By taking both parent–child tag relations and child–parent tag relations into consideration, we extract valuable features from the hidden latent representation of all the tag nodes. Specially, we apply Multi-Faceted Question GNN to tag-question relations so as to capture multiple facets of semantics from tags to questions. It is worth noting that the semantics of questions do not influence the semantics of tags, since the tags remain static in CQA websites. Also, there is no question–question relation in CQA websites. Instead, users should seek related questions by employing search mechanisms based on relevant tags. Then, we use Tri-Relational Question-Tag GNN for message passing within the graph. Specially, we introduce the concept of Multi-Faceted Question GNN, designed to capture multiple facets of semantics so as to tackle **Challenge 2**. As depicted in Fig. 2(b), in our proposed method, we encode each question into multi-faceted features. This approach allows for greater flexibility and provides a higher likelihood of extracting semantics from relevant tags. Through these multiple facets, we capture multiple domain features for the question, derived from relevant tags. In Fig. 2(b), the question acquires domain features such as 'Delicacy' and 'Trivia', distinctly disparate from the domain feature 'History'. This approach allows for a more comprehensive representation of the question, encompassing various facets and their corresponding semantic meanings.

For evaluation, we conduct comprehensive experiments to verify the effectiveness of our model. The performance comparison with the state-of-the-art methods confirms the superiority of our proposed model on three real-world datasets for automatic question tagging.

In this paper, our main contributions can be summarized as fourfold:

- We propose a Tri-Relational Multi-Faceted Graph Neural Network, namely TRMFG, to effectively perform the question tagging task in CQA websites. It generates Tri-Relational Question-Tag Graph and Tri-Relational Question-Tag GNN that help capture the underlying relations between questions and tags.
- We design a Tri-Relational Question-Tag Graph to model the questions, tags, and relations in CQA websites. By analyzing the characteristics of CQA websites, we construct the questions and tags as a heterogeneous graph for further study.
- We design a Tri-Relational Question-Tag GNN to learn informative node features for questions and tags in CQA websites. Specially, we design a Multi-Faceted Question GNN, which serves to extract multiple facets of semantics from related tags for questions. Also, the whole Tri-Relational Question-Tag GNN does message passing and learns latent representations of questions and tags.
- We validate the effectiveness of our model using three real-world datasets in experiments, i.e., Zhihu, Stack Overflow, and Zhuanzhi. These datasets are obtained from authentic CQA websites. Comprehensive experiments demonstrate the effectiveness of our model compared to state-of-the-art methods for automatic question tagging.

2. Related work

In this section, we provide an overview of the relevant literature associated with our framework. We introduce the work about question tagging tasks and graph neural networks.

2.1. Question tagging

AQT holds significance across various functions such as recommendation systems [5–7], expert-finding systems [8–10], and search engines [11–13] within the context of CQA websites.

In fields of recommendation systems, tags play an important role in the extraction of essential question information, facilitating the provision of appropriate answers for questions [20–23]. Recommendation system platforms enhance repository organization through tag-based systems, offering users comprehensive suggestions to optimize user experience and content accessibility. Izadi et al. [24] have proposed work for tagging software projects of recommendation system platforms. They have found that implementing appropriate tags significantly increases repository visibility on these platforms, thereby affirming the advantageous role of AQT in enhancing CQA tasks. Moreover, in expert-finding systems, which demand access to extensive information, these question-specific tags empower users to determine whether the retrieved information aligns with their specific requirements [25–28]. Interactions within expert-finding systems involve highly specialized questions and answers, often necessitating additional illumination for comprehension. Employing tags for questions within these systems will facilitate the extraction of key information, thereby enhancing the efficiency of information retrieval processes. Costa et al. [29] have devised approaches focusing on tags to make expert recommendations. Through the strategic tagging of questions and identification of the greatest expertise associated with these tags, their methodologies work in expert recommendation systems. When it comes to search engines, tags enable users to quickly find the information they need [13,30,31]. The vast and fragmented nature of information on Internet poses challenges for search engines in generating precise results. However, by leveraging tags, search engines can significantly improve their ability to pinpoint relevant information, enhancing the persuasiveness and relevance of search results for users. Tsai et al. [32] have extracted multiple object tag information from images, enabling users to perform image searches that yielded better results in comparison to those common search engines. Hence, there is significant value in comprehending and exploring AQT within CQA websites.

Numerous scholarly works have extensively examined various tagging methods that effectively integrate supplementary information. For example, simplified GCNs [33] capture complex transitions of items with session sequences to fulfill the tagging tasks. Likely, Wang et al. [34] have explored the construction of multi-relation edges between items, resulting in a more comprehensive understanding of users' sequential behaviors. Knowledge graphs can also help tag texts. Wang et al. [35] built the relation graphs by knowledge graph and apply attention mechanism to the model. These methods enable the model to effectively tag long-form text, which can be treated as extended questions.

In recent years, researchers tempted to focus more on the question tagging tasks themselves rather than on common text classification tasks. These question-tagging methods mostly depend on graph-based structures. Nie et al. [18] can be regarded as the first batch of researchers to utilize the novel topic to tag questions in the CQA websites. They constructed a Directed Acyclic Graph (DAG) for the tags to transfer knowledge by regularizing their hierarchical relations. Then the model cannot only learn tag and question embedding but recommend the tags to questions based on the embedding interaction as well. However, the state-of-the-art method, PROFIT, only concentrates on the semantics of the question, while ignoring the user-assigned tags which can provide complementary information for the question. With hierarchical learning taxonomy [14], questions can be searched based on their chapters or topics. Also, Zhang et al. [19] presented a tag ranking model called HERE so as to tag the questions with unseen tags. Through the incorporation of a DAG-based information propagation module, their proposed framework enables the effective transmission of meaningful information from parent tags to their respective child tags. Therefore, the unseen tags in the tag database can be better represented. The state-of-the-art method HERE [19] mainly concentrates on relations from parent tag nodes to child tag nodes, neglecting the pivotal roles of mutual information propagation. Furthermore, it only rewards positive question-tag pairs instead of fully utilizing them.

Different from the above methods, we propose a novel automatic question tagging method to sufficiently exploit the semantics of user-assigned tags and different relations between the question and tags. Specifically, we construct a Tri-Relational Question-Tag Graph to model relations among the input question, the user-assigned tags, and the candidate tags. Then, we propose a Tri-Relational Question-Tag GNN to learn informative node features for questions and tags, improving the effectiveness of the semantic match between the input question and candidate tags.

2.2. Graph Neural Networks

Recently, Graph Neural Networks (GNNs) have shown promising abilities to analyze graph-structured data [36–39]. In social networks [40], the data often exhibits a natural structure that can be effectively represented as graph data structures. Consequently, researchers have extensively explored various graph neural network architectures to represent different relations in CQA websites.

After the convolutional neural networks (CNNs) are applied to graph-structured data, GCN [41] models perform well. Defferrard et al. [42] managed to generalize the CNNs directly to graphs. Their proposed technique offered the same linear computational complexity and constant learning complexity as classical CNNs, being universal to any graph structure. Wu et al. [33] simplified the GCN model that it can convert large filters to small ones. With the explosive growth

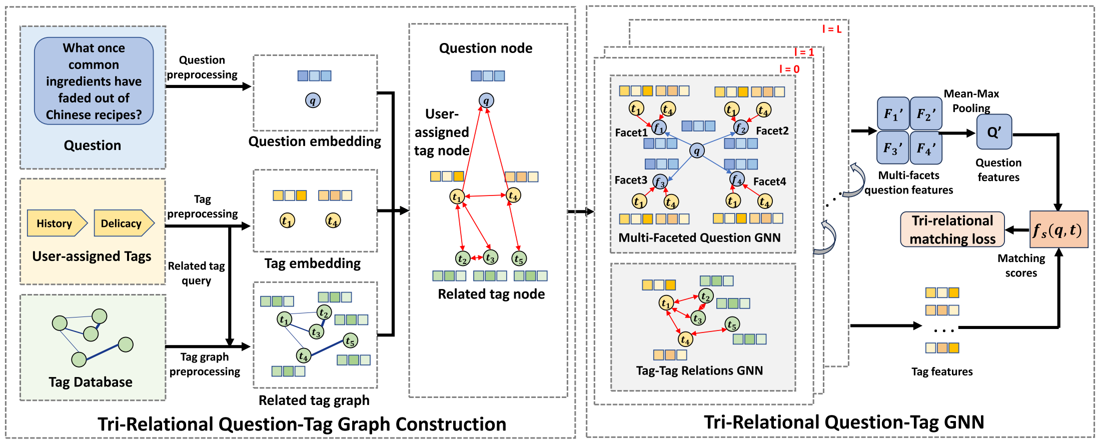


Fig. 3. The framework of TRMFG: (1) We preprocess the question, user-assigned tags and tag database to get question embedding, tag embedding and related tag graph; (2) we construct **Tri-relational Question-Tag Graph** to model the three types of relations within the CQT framework; (3) we propose **Tri-relational Question-Tag GNN** to do the message passing between questions and tags. Particularly, we design the **Multi-Faceted Question GNN** to encode questions into multiple facets and extract diverse semantics from tags to questions. Also, we design **Tag-Tag Relations GNN** for message passing between tags; (4) Finally, we propose Tri-Relational Matching Loss Function to optimize our model.

of social network data, GraphSAGE [43] is the first to attempt to do node-wise sampling coupled so that node embedding can be trained in mini-batches.

After the attention mechanism [44] comes into the vision of researchers, it has been widely used to decode the data so that the most relevant vectors will get higher weights. Therefore, Velickovic et al. [45] manage to integrate this idea into GCNs. When aggregating the information, the neighborhood will acquire different weights for each node. Brody et al. [46] put forward a new version of GAT, which change the weights into dynamic to get better attention results.

While social networks may have different types of entities and relations, the social graphs consist of different nodes and edges, which can be regarded as heterogeneous graphs. Chen et al. [47] projected metric embedding to heterogeneous graphs to capture first-order and second-order proximities. In HeGAN [48], Hu et al. sampled the heterogeneous graphs more effectively and efficiently and captures the rich semantics. The variants of the GCN models help better filter the information and adapt to various scenarios. NGCF [49] is the first try to design GCN under collaborative filtering settings. It deals with a user–item bipartite graph and do the recommendation. Then, He improves the NGCF model and simplifies the layers, which is called LightGCN [50]. GMCF [51] distinguishes the inner interactions and cross interactions to exploit the information carried by the different interactions. The denoising module consists of hard and soft denoising strategies. SGC [33] also makes the NGCF model simpler by removing nonlinearities. Li et al. [52] also improve the GCN-based CF models by proposing SGGCF to capture the high-order interactions. ITSM-GCN [53] is designed to sample positive training data. By the similarity-based samplers and score-based samplers, it outperforms other models.

In our work, we model the questions and tags in graph structure and apply the graph neural networks to aggregate information under the framework of graph neural networks.

3. Problem formulation

We introduce the notations used in this paper. For questions and tags in CQA websites, they can be represented as a heterogeneous graph $G = (V, E, M)$, where $V = V_q \cup V_t$ is the union of question nodes set $V_q = \{q_1, \dots, q_k\}$ and tag nodes set $V_t = \{t_1, \dots, t_n\}$. k is the number of questions and n is the number of tags. The question and tag nodes within the graph are both products of embedding features, originating from the questions and tags sourced from CQA websites. We present the preprocessing method in Section 5.1 to analyze the details. $E = E_{t-q} \cup E_{t-t}$ is the set of edges that represents the relations between nodes. $E_{t-q} = \{(t, e, q) | t \in V_t, e \in M_{t-q}, q \in V_q\}$ is the set of edges between user-assigned tags and corresponding questions, where M_{t-q} is the set of question-tag relations. $E_{t-t} = \{(st, e, tt) | st \in V_t, e \in M_{t-t}, tt \in V_t\}$ is the set of edges between tags and tags in the tag database, where M_{t-t} is the set of tag-tag relations.

The core of our TRMFG model is to learn a retrieval function: $\mathbb{T} = F((q, t); q, E_{t,q})$, which takes the question along with the user-assigned tags as the input and outputs the retrieved tags for the question. By learning the question and tag features by our model, we calculate the similarity score between each tag and the question, and get the tags with higher scores for the question as the retrieved tags.

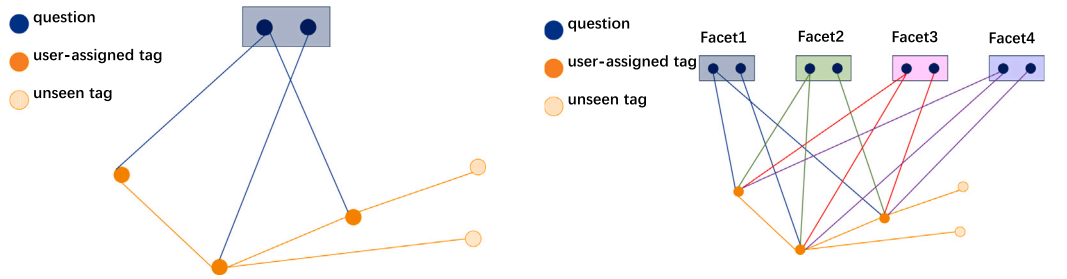
The main notations of this work are summarized in Table 1.

Table 1
The main symbols and their meaning in TRMFG.

Notation	Definition
$G = (V, E, M)$	Question-tag graph
V	Node set
E	Edge set
M	Relation set
$\mathcal{Q}$	The original question embedding
$\mathcal{T}$	The original tag embedding
W_j	Trainable matrix of a linear projection
b_j	Trainable bias for question embedding
F_j	Question facet
p	Number of facets
(t, e, q)	Tag-question edge, where t is tag node and q is question node
(st, e, tt)	Tag-tag edge, where st is source tag node and tt is target tag node
K^i	K-linear projection
W_e^{att}	Matrix for edge
Q^i	Q-linear projection
μ_e	Prior tensor
$G^{(l)}$	Updated embedding layer
$\mathcal{Q}'$	The final feature of question embedding
$\mathcal{T}'$	The final feature of tag embedding
$\mathcal{L}$	Loss function

4. Methodology

In this section, we present Tri-Relational Multi-Faceted Graph Neural Networks (TRMFG) for Automatic Question Tagging. The architecture of our model is illustrated in Fig. 3, which consists of **Tri-Relational Question-Tag Graph** and **Tri-Relational Question-Tag GNN**, which is composed of **Multi-Faceted Question GNN** and **Tag-Tag Relations GNN**. The Tri-Relational Question-Tag Graph models the relations between parent tags and child tags, child tags and parent tags, as well as tags and questions. Each of these relations serves the purpose of distributing information between the corresponding nodes.



(a) Common tri-relational graph structure. (b) Tri-relational graph structure with multiple facets questions.

Fig. 4. Examples of tri-relational graph structure. (a) shows that in common tri-relational graph structures, questions already have relations with user-assigned tags, while they have bidirectional relations with unseen tags. (b) shows that in our models, we encode questions into multiple facets, while conserving their relations with the user-assigned tags.

Then we develop the Tri-Relational Question-Tag GNN for message passing between the relations. In particular, we propose Multi-Faceted Question GNN for tag-question relations. By encoding questions into diverse vectors, with each vector designed to emphasize different potential tags. This approach facilitates a more comprehensive representation of the underlying meaning embedded within the questions. Through the process of message passing, we get the extraction of multiple facets of semantics for each question.

In the final stage of retrieving proper tags for questions, we employ a scoring mechanism to evaluate tags in relation to the different facets of questions. Individual scores are calculated for tags in comparison to each question facet, and these scores are recorded. The highest scoring matches are then identified based on these scores.

4.1. Tri-relational Question-Tag Graph

In our question tagging model, we construct a Tri-Relational Graph Structure to effectively represent the three key relations in CQA websites. These relations play a significant role in our CQT framework and are crucial for extracting the hidden latent representation of questions and tags. The first relation is the parent–child tag relation, which transmits global features from parent tag nodes to child tag nodes, providing them with shared semantic information. Conversely, the child–parent tag relation transmits side features from child tag nodes to parent tag nodes. Lastly, the tag-question relation establishes unidirectional connections from tag nodes to question nodes. Through these types of relations, questions can acquire semantics and relevant information from user-assigned tags.

However, we do not take question-tag relations into consideration. Since the tags in CQA websites remain static, extracting semantics from questions to tags becomes unattainable within this framework. It is worth noting that CQA websites lack question–question relations. Instead, users should seek related questions by employing search mechanisms based on relevant tags. This enables them to discover additional questions related to the same tags.

Additionally, we preprocess the questions and tags to obtain question and tag embeddings. The preprocessing method is discussed in Section 5.1. As shown in Fig. 4(a), the common tri-relational question graph we construct in our AQT model contains bidirectional relations between parent and child tag nodes, along with unidirectional relations extending from tags to questions. The user-assigned tags may have relations with unseen tags, which are in the database but not assigned to any question.

With the objective of enabling our model to capture multiple facets of semantics from tags, we employ the Multi-Faceted Question GNN discussed in Section 4.2.1 to effectively encode the questions. The Fig. 4(b) shows the tri-relational graph structure with multiple facets of questions. Meanwhile, the tags maintain their connections with the original questions. The multiple facets will not disturb the question-tag graph we build.

4.2. Tri-relational Question-Tag GNN

After we develop the Tri-Relational Graph Structure to represent the three key relations in CQA websites, we propose the Tri-Relational Question-Tag GNN to learn the hidden node representation.

4.2.1. Multi-Faceted Question GNN

Questions in CQA websites may have user-assigned tags. We try to do GNN message passing from user-assigned tags to the question, so the question can generate more comprehensive representations from user-assigned tags and retrieve multiple semantics. Therefore, we try to capture the diverse facets of questions by considering the contribution of different tags to different facets.

We utilize $Q = \{q_i \in \mathbb{R}^m\}_{i=1}^k$ from question nodes set V_q to represent the original question embedding, where k is the number of questions and m is the dimension of question embedding. To capture the diverse facets of questions, we encode the origin questions into multiple vectors.

$$\underset{j=1,\dots,p}{F_j} = QW_j + b_j \in \mathbb{R}^m, \tag{1}$$

where p is the number of facets. $W_j \in \mathbb{R}^{m\times m}$ is a trainable matrix of a linear projection. Each question feature gets a trainable bias $b_j \in \mathbb{R}^m$ to approach to related tags and enables the acquisition of semantics from that particular tag. This approach ensures that a broader range of question facets is captured and represented.

Then, for each question facet F_j, we hope to acquire the semantics from related tags. We first calculate different attention weights of each tag-question edge $(t,e,q) \in M_{t-q}$:

$$\mathrm{Att}^i_{F_j}(t,e,q) = (K^i(t)W_e^{att}Q^i(q)^T)\cdot\frac{\mu_e}{\sqrt{m}} \in \mathbb{R}, \tag{2}$$

where t is the source tag node and q is the target question node, and e is the edge from source tag node to target question node. $Att^i_{F_j}(t,e,q)$ represents the i_{th} attention head of the edge. $K^i(t)$ is a linear projection: $\mathbb{R}^m \to \mathbb{R}^{\frac{m}{h}}$ that projects tag to the key vector, where h is the number of attention heads. $Q^i(q)$ is a linear projection that projects a question to a query vector. The $W_e^{att} \in \mathbb{R}^{\frac{m}{h}} \times \mathbb{R}^{\frac{m}{h}}$ is the matrix for tag-question edge to calculate the dot product between the query and key vector. And $\mu_e \in \mathbb{R}^{|e|}$ is a prior tensor to denote the general significance of each tag-question relation, serving as an adaptive scaling to the attention.

Then, we concatenate all the attention heads together to evaluate the importance of the source tags set $N(q)$:

$$\mathrm{Att}_{F_j}(t,e,q) = \underset{\forall t\in N(q)}{\mathrm{Softmax}}\Big(\underset{i=1,\dots,h}{\Big\Vert}\mathrm{ATT}^i_{F_j}(t,e,q)\Big) \in \mathbb{R}^h, \tag{3}$$

where $\Vert$ represents concatenation, and h is the number of heads for the edge. When we gather all the attention from its user-assigned tag nodes $N(q)$ for each q, $\mathrm{Softmax}$ fulfills $\sum_{\forall t\in N(q)} Att_{F_j}(t,e,q) = \mathbb{1} \in \mathbb{R}^h$.

Meanwhile, we manage to pass the message from user-assigned tag nodes to the question node. In the tag-question edge, the message propagation can be formulated as follows:

$$\mathrm{Msg}^i_{F_j}(t,e,q) = \mathrm{M\_Linear}^i_t(G^{(l-1)}[t])W_e^{msg} \in \mathbb{R}^{\frac{m}{h}}, \tag{4}$$

$$\mathrm{Msg}_{F_j}(t,e,q) = \underset{i=1,\dots,h}{\|} \mathrm{Msg}^i_{F_j}(t,e,q) \in \mathbb{R}^{h\cdot\frac{m}{h}}, \tag{5}$$

where the $\mathrm{M\_Linear}^i_t$ is the linear projection: $\mathbb{R}^m \rightarrow \mathbb{R}^{\frac{m}{h}}$ that projects the tags into the tag-question messages, and h is the number of heads for the edge. $G^{(l-1)}[t]$ is present tag embedding layer. The $W_e^{msg} \in \mathbb{R}^{\frac{m}{h}} \times \mathbb{R}^{\frac{m}{h}}$ is the matrix to incorporate the edge dependency. $\|$ represents concatenation of all the messages for the (t,e,q) edge.

In our model, the updated vector of the target question embedding in F_j is calculated using the following formula:

$$\tilde{Q}_{F_j} = \underset{i=1,\dots,h}{\|} \Big(\sum_{t\in N(q)} \mathrm{Att}^i_{F_j}(t,e,q) \cdot \mathrm{Msg}^i_{F_j}(t,e,q) \Big) \in \mathbb{R}^m, \tag{6}$$

where the $\tilde{Q}_{F_j}$ is the update feature of F_j and $N(q)$ is the set of source tags. We concatenate all the information from all the source tag nodes.

Then we map the vectors back to the former layer to get the output of l_{th} question layer of j_{th} facets:

$$G^{(l)}_{F_j}[q] = \tilde{Q}_{F_j} + G^{(l-1)}_{F_j}[q], \tag{7}$$

where $G^{(l)}$ is the updated embedding layer.

And we set L to be the final layer of question embedding and get the j_{th} facets of question embedding:

$$Q'_{F_j} = G^{(L)}_{F_j}[q] \in \mathbb{R}^m, \tag{8}$$

where Q'_{F_j} is the final feature of the j_{th} facets of question embedding.

4.2.2. Tag-tag relations GNN

In the tri-relational question-tag graph, message passing for all tag nodes is bi-directional. Parent nodes provide global features to their child nodes, while child nodes transmit side features to their parent nodes. For each tag node within the question-tag graph, we gather hidden latent representations from all the connected tag nodes, regardless of whether they are parent nodes or child nodes. We utilize $\mathcal{T} = \{t_i \in \mathbb{R}^m\}^n_{i=1}$ from tag nodes set V_t to represent the original tag embedding, where n is the number of tags and m is the dimension of tag embedding.

Firstly, we aim to compute its attention vectors of concerning tag-tag edge $(st,e,tt) \in M_{t-t}$:

$$\mathrm{Att}^i(st,e,tt) = (K^i(st)W_e^{att}Q^i(tt)^T) \cdot \frac{\mu_e}{\sqrt{m}} \in \mathbb{R}, \tag{9}$$

where st is the source tag node and tt is the target tag node from $\mathcal{T}^m$. e is the edge between source tag node and target tag node. $Att^i(st,e,tt)$ represents the i_{th} attention head of the tag-tag edge. $K^i(t)$ is the linear projection: $\mathbb{R}^m \rightarrow \mathbb{R}^{\frac{m}{h}}$. It projects the source tag to the key vector. And the $Q^i(q)$ is the linear projection that projects the target tag to query vector of the attention mechanism. $W_e^{att} \in \mathbb{R}^{\frac{m}{h}} \times \mathbb{R}^{\frac{m}{h}}$ is the matrix for tag-tag edge to calculate the dot product between the query and key vector. $\mu_e \in \mathbb{R}^{|e|}$ is a prior tensor to denote the general significance of each tag-tag relation, serving as an adaptive scaling to the attention.

Then all the attention heads are concatenated for evaluating importance of the source tag.

$$\mathrm{Att}(st,e,tt) = \underset{\forall st\in N(t)}{\mathrm{Softmax}}\big(\underset{i=1,\dots,h}{\|} \mathrm{Att}^i(st,e,tt)\big) \in \mathbb{R}^h, \tag{10}$$

where $\|$ represents concatenation, and h is the number of heads for the edge. When gathering all the attention from its source tag nodes $N(tt)$ for each target tag node tt, Softmax fulfills $\sum_{\forall st\in N(t)} Att(st,e,tt) = 1_{h\times 1}$.

Meanwhile, when evaluating the importance of source tag node, we transmit the messages to target tag node. The message passing of tag-tag edge is:

$$\mathrm{Msg}^i(st,e,tt) = \mathrm{M\_Linear}^i_{st}(G^{(l-1)}[t])W_e^{msg} \in \mathbb{R}^{\frac{m}{h}}, \tag{11}$$

$$\mathrm{Msg}(st,e,tt) = \underset{i=1,\dots,h}{\|} \mathrm{Msg}^i(st,e,tt) \in \mathbb{R}^{h\cdot\frac{m}{h}}, \tag{12}$$

where the $\mathrm{M\_Linear}^i_{st}$ is the linear: $\mathbb{R}^m \rightarrow \mathbb{R}^{\frac{m}{h}}$ that projects source tag nodes into the i_{th} messages, and $W_e^{msg} \in \mathbb{R}^{\frac{m}{h}} \times \mathbb{R}^{\frac{m}{h}}$ is the matrix to incorporate the edge dependency. $G^{(l-1)}[t]$ is present tag embedding layer. $\|$ represents concatenation for all the messages from all the source tag nodes.

Then, we update the feature of tag embedding by aggregating all the source tag nodes with their attention weights.

$$\tilde{T} = \underset{i=1,\dots,h}{\|} \Big(\sum_{st\in N(tt)} \mathrm{Att}^i(st,e,tt) \cdot \mathrm{Msg}^i(st,e,tt) \Big) \in \mathbb{R}^m, \tag{13}$$

where the $\tilde{T}$ is the update feature of $\mathcal{T}$ and $N(tt)$ is the set of source tags.

We map the feature back to the former layer to get the output of l_{th} tag layer $G^{(l)}$:

$$G^{(l)}[t] = \tilde{T} + G^{(l-1)}[t], \tag{14}$$

where $G^{(l)}$ is the updated embedding layer. Then we set L to be the final layer of tag embedding.

$$T' = G^{(L)}[t] \in \mathbb{R}^m, \tag{15}$$

where T' is the final tag features. After acquiring the hidden latent representations of various facets, we will employ these outputs to predict the relations between questions and tags.

4.3. Tri-relational matching loss

Due to the diverse facets of question semantics, we implement a mean-max pooling component in our design to keep related representations close. Different semantics from different tags may be unrelated. In cases where semantics are quite different, we hope to identify the **unique** characteristics of each question. Therefore, we apply the max pooling method to get these semantics.

$$Q'_{max} = \underset{j=1,\dots,p}{\mathrm{MaxPooling}}(Q'_{F_j}), \tag{16}$$

where $(G^{(l)}_{F_j}[q])$ is the j_{th} facets of final question embedding. Max-Pooling focuses on the side semantics and helps us get the most related semantics of questions.

However, it is important to consider that in some cases, tag semantics may be interconnected, given that the tags associated with questions are interrelated. Therefore, acquiring a comprehensive understanding of the **global** semantics, encompassing all facets of the questions, becomes crucial. To achieve this, we employ the Mean-Pooling method.

$$Q'_{mean} = \underset{j=1,\dots,p}{\mathrm{MeanPooling}}(Q'_{F_j}), \tag{17}$$

where $(G^{(l)}_{F_j}[q])$ is the j_{th} facets of final question embedding.

Then we combine the Max-Pooling results and Mean-Pooling results together to preserve better information.

$$Q' = \|(Q'_{mean}, Q'_{max}), \tag{18}$$

where $\|$ represents concatenation. The Eq. (16) shows the mean-max pooling method to capture potential semantics.

When evaluating the loss function for the question tagging task, it is crucial to consider that each question may be related to only a limited number of tags. Consequently, while conducting model sampling, it should maintain an abundant proportion of negative and positive samples, ideally numbering one hundred or more.

We refer to the study of Hu et al. [54] to sample the negative sets and calculate the loss. To better maximize the mutual information between questions and their predicted tags, we try to sample the set of negative tags from tag nodes set V_t with uniform distribution. The loss function of the tri-relational GNN $\mathcal{L}_{tr}$ can be defined as the following formula:

$$\mathcal{L}_{tr} = -\sum_{(q_i,t_j)\in E} \mathbb{E}_{\{t_k\}\sim\{p(t)\}} log \frac{e^{f_s(q_i,t_j)}}{e^{f_s(q_i,t_j)} + \sum_{t_k} e^{f_s(q_i,t_k)}}, \tag{19}$$

where $f_s(q_i,t_j)$ computes the dot product between question $q_i \in Q'$ and tag $t_j \in T'$. The $\mathbb{E}_{\{t_k\}\sim\{p(t)\}}$ samples a set of negative tags vertices from tag nodes set V_t with uniform distribution.

Also, we use L2 normalization for the Tri-Relational GNN. When calculating the L2 loss $\mathcal{L}_{L2}$, we set the weight Ψ_{L2} to be 10^{-4}. Then, the tri-relational matching loss for our TRMFG model is:

$$\mathcal{L}_{trg} = \mathcal{L}_{tr} + \Psi_{L2}\mathcal{L}_{L2} \tag{20}$$

4.4. Multiple matching component

During the TRMFG model evaluation process, we employ data obtained from CQA websites. It is worth noting that each question is treated as an independent entity since there is no information propagation between questions in our proposed model. It enables the model to retrieve related tags for a new input question without any redundant interactions with other questions, thereby enabling real-time inference.

Considering the diverse facets of question semantics, the challenge lies in effectively retrieving appropriate tags that suit each question. To address this, we utilize a dot product score to measure the similarity between tags and each semantic facet of the question. This enables us to match the most suitable tags to the questions. The dot product score function is supposed to be:

$$s(q,t) = f_s(q,t) \tag{21}$$

Then we use the function to calculate all the scores for each question and tag:

$$\mathbb{S}_j = \{s(q,t) | t \in T', q \in Q'_{F_j}\}, \tag{22}$$

where $\mathbb{S}_j$ is the scores set of the j_{th} facets of questions and tags.

Finally, we compare all the scores of questions in each facet and tags, then reserve the tags with the highest scores for each question:

$$\mathbb{T} = \{t = argmax(\mathbb{S}_1,\dots,\mathbb{S}_p)\}, \tag{23}$$

where p is the number of facets. The $\mathbb{T}$ is the set of predicted tags. We finally retrieve the proper tags for each question.

5. Experimental results

To investigate the effectiveness of our proposed method, we conduct extensive experiments on three datasets.

5.1. Experimental setup

Datasets: We evaluate our model on three datasets: Zhihu, Stack Overflow, and Zhuanzhi. The Stack Overflow dataset is created based on a public dataset on StackLite.[3] Since the original dataset lacks tag-tag relations, we filter the tags and reconstruct the dataset.

The **Zhihu** dataset is compiled and derived from Zhihu website, encompassing a collection of published questions along with their related tags. The questions and tags cover almost all fields in our daily life. We have obtained data from the website before February 2022. To create this dataset, the RoBERTa-base model[4] is utilized to generate embeddings for both the questions and tags. The **Stack Overflow** dataset contains question IDs and question-tag relations. We use GloVe[5] to embed the tags. Then we build the tag-tag relations based on the embedding similarity and consider the relations with scores above the threshold. The **Zhuanzhi** dataset closely resembles the Zhihu dataset. We collect the published questions and their corresponding tags from the Zhuanzhi website before February 2022. Embeddings are generated for both the questions and tags in a manner similar to the Zhihu dataset.

Table 2
Statistics of three datasets.

Dataset	# Questions	# Tags	# Q-T	# T-T
Zhihu	30 102	73 285	108 258	133 522
Stack overflow	8141	10 976	31 865	18 761
Zhuanzhi	4987	5603	10 466	6335

Polysemy, homonymy and misspelled words within question texts have the potential to introduce disruptions in comprehending the semantics of the questions. To relieve these problems, we adopt RoBERTa [55] pretrained on large-scale Internet corpus to encode questions, which facilitates the robust comprehension of polysemy, homonymy, and misspelled words in a context-aware manner. The RoBERTa model is to comprehend the overall semantics of questions, while our proposed method is designed to extract fine-grained semantics from the tags associated with these questions. Focusing on the automatic question tagging task in this paper, we do not include the typo-fixing procedure in our experiments for all compared methods.

The statistics of the adopted datasets are summarized in Table 2.

Setup: We divide the question-tag relations into three sets: training, validation, and testing. For each dataset, 50% of the tag nodes have been allocated as seen tags, while the remaining 50% are categorized as unseen tags. Then half of the question-tag relations associated with seen tags are combined with all those originating from the unseen tags to form the test set. Following this partitioning, 10% of the remaining question-tag relations associated with the seen tags are pointed as the validation set, serving the crucial purpose of fine-tuning our model. Subsequently, the remaining question-tag relations are allocated to the training set.

In our experimental setup, we ensure that the three sets: training, validation, and testing maintain complete independence from each other. This partitioning strategy is crucial. It enables us to accurately assess the model's capability to proficiently address unseen tags within the context of CQA websites.

Metrics: For retrieval tasks, researchers usually adopt two metrics [56]: **Recall** score and **Normalized Discounted Cumulative Gain (NDCG)** score. We also adopt **Precision** score to evaluate the TRMFG model. In our automatic question tagging task, the Precision score assesses the proportion of retrieved tags that are successfully predicted out of all the retrieved tags. Recall score assesses the proportion of retrieved tags that are successfully predicted out of all the actual relevant tags. Normalized Discounted Cumulative Gain (NDCG) score assesses the quality of the retrieved tags by considering the relative positions of the true relevant tags in the retrieved tag list. By calculating the Discounted cumulative gain (DCG) and Ideal DCG (IDCG) of the retrieved tags list, we can get the NDCG scores. By assessing the performance of the compared models using Precision, Recall, and NDCG scores, we can ascertain the overall effectiveness of the automatic question-tagging task.

[3] https://www.kaggle.com/datasets/stackoverflow/stacklite

[4] https://huggingface.co/roberta-base

[5] https://nlp.stanford.edu/projects/glove/

Table 3
The question tagging results on three datasets. The bold indicates the best scores.

Datasets	RESULT	GCN	GAT	APPNP	RGCN	HGT	HERE	PROFIT	TRMFG (ours)
Zhihu	PRECISION@5	0.005	0.002	0.043	0.008	0.068	0.025	0.011	**0.092**
	PRECISION@10	0.004	0.002	0.037	0.008	0.049	0.018	0.008	**0.064**
	PRECISION@15	0.004	0.003	0.032	0.007	0.040	0.015	0.007	**0.050**
	RECALL@5	0.012	0.002	0.082	0.023	0.137	0.086	0.032	**0.173**
	RECALL@10	0.016	0.004	0.138	0.041	0.193	0.131	0.051	**0.239**
	RECALL@15	0.022	0.009	0.182	0.060	0.232	0.164	0.068	**0.281**
	NDCG@5	0.014	0.002	0.069	0.018	0.122	0.074	0.029	**0.158**
	NDCG@10	0.016	0.003	0.092	0.026	0.146	0.093	0.037	**0.187**
	NDCG@15	0.018	0.004	0.108	0.032	0.159	0.105	0.043	**0.202**
Stack overflow	PRECISION@5	0.039	0.042	0.043	0.006	0.018	0.028	0.038	**0.045**
	PRECISION@10	0.024	0.026	0.025	0.005	0.011	0.019	0.024	**0.027**
	PRECISION@15	0.018	0.019	0.019	0.004	0.009	0.015	0.018	**0.020**
	RECALL@5	0.069	0.074	0.072	0.011	0.076	0.050	0.071	**0.083**
	RECALL@10	0.084	0.090	0.082	0.015	0.096	0.068	0.092	**0.104**
	RECALL@15	0.091	0.101	0.092	0.017	0.111	0.079	0.104	**0.119**
	NDCG@5	0.066	0.077	0.080	0.011	0.071	0.049	0.070	**0.088**
	NDCG@10	0.074	0.084	0.084	0.013	0.078	0.056	0.080	**0.094**
	NDCG@15	0.076	0.088	0.088	0.013	0.084	0.061	0.085	**0.100**
Zhuanzhi	PRECISION@5	0.002	0.010	0.011	0.008	0.008	0.013	0.010	**0.015**
	PRECISION@10	0.001	0.006	0.010	0.008	0.009	0.013	0.009	**0.017**
	PRECISION@15	0.006	0.004	0.011	0.009	0.010	0.012	0.008	**0.013**
	RECALL@5	0.001	0.036	0.038	0.018	0.011	0.034	0.003	**0.043**
	RECALL@10	0.004	0.041	0.049	0.033	0.027	0.045	0.034	**0.072**
	RECALL@15	0.005	0.054	0.062	0.045	0.039	0.079	0.075	**0.083**
	NDCG@5	0.001	0.027	0.036	0.013	0.009	0.022	0.003	**0.042**
	NDCG@10	0.001	0.029	0.037	0.019	0.012	0.029	0.014	**0.063**
	NDCG@15	0.002	0.033	0.039	0.023	0.017	0.039	0.027	**0.072**

5.2. Implementation details

We randomly hold 10% of the training set to form the validation set. By referring to the hyperparameters in original papers of baselines and utilizing grid-search, we cross-validate the hyperparameters of all experimented methods. In the question tagging task, the three datasets comprise thousands of question and tag nodes organized within a heterogeneous graph. This characteristic makes the data large-scale, considering the significant number of interconnected nodes and their complex relationships. We employ Adam optimizer as our choice of optimization algorithm while training the model, and the learning rate was set to 2e−3. Additionally, we utilized a gamma value of 0.99 in the optimization process. For all the datasets, we set the batch size as 4000.

For the parameters in the model, we set the layer number of TRMFG as 2, and the facets number as 4. The head number of edges is set to 4. The output embedding dimension is consistent with the corresponding original embedding dimension.

Our experiments are conducted on the Linux operating system: Ubuntu 20.04.1 OS. We employ the NVIDIA RTX 3090 for training our model and the PyTorch framework for code development. Furthermore, we utilize the Conda environment management system to execute our codes.

Our data[6] and code[7] are publicly available.

5.3. Baselines

We select 7 state-of-the-art methods for automatic question tagging as our baselines:

1. **GCN** [41] is a deep convolutional network designed for graph-structured data. It is often used for node classification and link prediction tasks.
2. **GAT** [45] applies attention function to GCN. It allocates different weights to each neighbor node, then the more significant neighbor node can be distinguished.
3. **APPNP** [57] associates PageRank with GCN. Referring to the node propagation mode of PageRank, an improved adjacent feature propagation mode is proposed.
4. **RGCN** [58] is a simple attempt of GCN on heterogeneous graph. It aims at solving problems on heterogeneous graphs that have heterogeneous nodes.
5. **HGT** [17] makes full use of the attribute information of the heterogeneous graph. It comes up with the idea of sharing the parameters for better generalization.
6. **HERE** [19] is a graph-guided topic ranking model. It tags questions in CQA websites.
7. **PROFIT** [18] is an end-to-end interactive embedding model to tag the questions. It learns the embedding of questions and tags by projecting them into the same space.

Among the baseline methods, **GCN**, **GAT**, and **APPNP** are isomorphic graph networks, while **RGCN** and **HGT** are heterogeneous graph models. **HERE** and **PROFIT** are question tagging methods in the real world. These GNN models focus on different relations of questions and tags to capture the hidden latent representation.

5.4. Results and discussion

We conduct experiments to investigate the effectiveness of our TRMFG model. For Precision, Recall, and NDCG scores, we set the number of retrieved tags to 5, 10, and 15 to identify the model's performance in retrieving tags and explore potential improvements associated with an increased number of retrieved tags. The results of all compared methods on three datasets are presented in Table 3.

From Table 3, we can get the following observations:

1. The graph neural networks perform poorly. Although they make use of different relations in CQA websites, they cannot capture or cannot capture semantics from user-assigned tags well.

[6] https://anonymous.4open.science/r/question-tagging-6D63/data/
[7] https://anonymous.4open.science/r/question-tagging-6D63/

Fig. 5. Crucial difference diagram of the Nemenyi test for Precision and Recall on three datasets and three numbers of retrieved tags.

2. Our proposed TRMFG model outperforms all baselines in the metrics. Compared with second-best score, our model achieves increase of 35.29%, 30.61%, 25.00%, 26.28%, 23.83%, 21.12%, 29.51%, 28.08%, 27.04% of PRECISION@5, PRECISION@10, PRECISION@15, RECALL@5, RECALL@10, RECALL@15, NDCG@5, NDCG@10, NDCG@15 on the Zhihu dataset, 4.65%, 9.21%, 8.33%, 7.21%, 10.00%, 11.90%, 13.64% on the Stack Overflow dataset and 15.38%, 13.16%, 60.00%, 5.06%, 16.67%, 70.27%, 84.62% on the Zhuanzhi dataset.
3. Compared with HERE, the-state-of-the-art method, our model achieves increases of 268.00%, 255.56%, 101.16%, 82.44%, 71.34%, 113.51%, 101.08%, 92.38% of PRECISION@5, PRECISION@10, PRECISION@15, RECALL@5, RECALL@10, RECALL@15, NDCG@5, NDCG@10, NDCG@15 on the Zhihu dataset, 60.71%, 42.11%, 33.33%, 66.00%, 52.94%, 50.63%, 79.59%, 67.86%, 63.93% on the Stack Overflow dataset and 15.38%, 30.77%, 8.33%, 26.47%, 60.00%, 5.06%, 90.91%, 117.24%, 84.62% on the Zhuanzhi dataset. The HERE model does not take child–parent tag relations into consideration, and it also does not acquire multiple semantics from user-assigned tags for questions.
4. Compared with PROFIT, the other the-state-of-the-art method, our model achieves increases of 736.36%, 700.00%, 614.29%, 440.63%, 368.62%, 313.24%, 444.83%, 405.41%, 369.77% of PRECISION@5, PRECISION@10, PRECISION@15, RECALL@5, RECALL@10, RECALL@15, NDCG@5, NDCG@10, NDCG@15 on the Zhihu dataset, 18.42%, 12.50%, 11.11%, 16.90%, 13.04%, 14.42%, 25.71%, 17.50%, 17.65% on the Stack Overflow dataset and 50.00%, 88.89%, 62.5%, 1333.33%, 111.76%, 10.67%, 1300.00%, 350.00%, 166.67% on the Zhuanzhi dataset. The PROFIT model focuses on parent–child relations more than child–parent tag relations, while it also does not acquire multiple semantics from user-assigned tags for questions.

Our model introduces a novel graph network aimed at enhancing feature learning. Within this framework, the Multi-Faceted Question GNN component enables the extraction of hidden latent representations from related tag nodes for selected question nodes. Furthermore, our approach introduces an innovative AQT method designed to improve the question-tagging method and provide valuable contributions to CQA websites. In contrast to the baseline models of GNN, our method demonstrates superior proficiency in learning node features. Additionally, we make full use of relations present in CQA websites, distinguishing our model from existing state-of-the-art models.

To verify whether our TRMFG model is significantly better than other methods, we adopt the Friedman test and Nemenyi test [59] to further compare the performance of TRMFG with the baselines. We perform the Friedman test at the 0.05 significance level under the null hypothesis which states that the performance of all algorithms is the same on all datasets and all metrics. We regard that the average ranks of all algorithms are equivalent.

The average ranks of TRMFG and baselines when using different evaluation metrics are summarized in Fig. 5. We can see that the null hypothesis is rejected on these two evaluation metrics. We also note that TRMFG performs better than baselines since the lower rank value is better. We also perform the Nemenyi test, which states that the performance levels of two algorithms are different if the corresponding average ranks differ by at least one critical difference (CD). Fig. 5 provides the CD diagrams, where the average rank of each algorithm is marked along the axis (lower ranks to the right). We observe that our TRMFG model achieves a comparable performance against APPNP, HGT, and HERE, and outperforms PROFIT, GAT, RGCN, and GCN.

5.5. Ablation experiments

In this section, we conduct ablation experiments to assess the effectiveness of the two important components in our model: Multi-Faceted Question GNN and Tri-Relational Question-Tag Graph. We show that removing any component from the model leads to a degradation in performance.

We compare our model with four variations:

- **w/o Multi-Faceted** The TRMFG model without the Multi-Faceted Question GNN.
- **w/o Q-T** The TRMFG model without Tag-Question Relations.
- **w/o C-P** The TRMFG model without Child–Parent Tag Relations.
- **w/o P-C** The TRMFG model without Parent–Child Tag Relations.

The results of these variations are presented in Table 4, providing evidence that the three relations within TRMFG are indeed functional. The impact of different relations on the performance is evident from our experiments. With no tag-question relations in the model, it can be regarded as a common isomorphic graph network and has poor performance. Upon removing the parent–child tag relations from the model, a notable decrease in performance is observed. What is more, the child–parent relations yield further enhancement in results due to the frequent usage of child tags in CQA contexts. Despite these improvements, the performance of them still falls short of the Tri-Relational GNN benchmark. Also, without the multiple facets, the performance is also poor. The results of the ablation experiments yield the following insights:

- **Multi-Faceted Question GNN** Multiple facets Question GNN plays a crucial role in enhancing the accuracy and overall effectiveness of our model in the question tagging task. By capturing the multiple facets of questions, it enables robust question-tag matching.
- **Tri-Relational Question-Tag Graph** The performance highlights the crucial role played by Tri-Relational Question-Tag Graph in capturing the relations between questions and tags. It leads to improved accuracy in addressing the question tagging challenge in CQA websites.

Table 4
The question tagging results w/o Multi-Faceted Question GNN and Three Relations. The bold indicates the best scores.

Dataset	Results	w/o Multi-Faceted	w/o T-Q	w/o C-P	w/o P-C	**TRMFG (ours)**
Zhihu	PRECISON@5	0.068	0.003	0.077	0.080	**0.092**
	PRECISION@10	0.049	0.002	0.052	0.051	**0.064**
	PRECISION@15	0.040	0.003	0.044	0.041	**0.050**
	RECALL@5	0.137	0.002	0.156	0.166	**0.173**
	RECALL@10	0.193	0.004	0.219	0.227	**0.239**
	RECALL@15	0.232	0.009	0.259	0.268	**0.281**
	NDCG@5	0.122	0.002	0.141	0.152	**0.158**
	NDCG@10	0.144	0.003	0.168	0.178	**0.187**
	NDCG@15	0.159	0.004	0.182	0.193	**0.202**
Stack overflow	PRECISON@5	0.018	0.043	0.040	0.038	**0.045**
	PRECISION@10	0.011	0.026	0.022	0.020	**0.027**
	PRECISION@15	0.009	0.019	0.016	0.017	**0.020**
	RECALL@5	0.076	0.074	0.043	0.050	**0.083**
	RECALL@10	0.096	0.090	0.056	0.066	**0.104**
	RECALL@15	0.111	0.101	0.067	0.076	**0.119**
	NDCG@5	0.071	0.077	0.042	0.048	**0.088**
	NDCG@10	0.078	0.084	0.049	0.056	**0.094**
	NDCG@15	0.084	0.088	0.052	0.060	**0.100**
Zhuanzhi	PRECISON@5	0.010	0.010	0.009	0.010	**0.015**
	PRECISION@10	0.009	0.008	0.008	0.010	**0.017**
	PRECISION@15	0.007	0.004	0.007	0.008	**0.013**
	RECALL@5	0.011	0.036	0.030	0.043	**0.043**
	RECALL@10	0.027	0.041	0.041	0.055	**0.072**
	RECALL@15	0.039	0.054	0.047	0.061	**0.083**
	NDCG@5	0.009	0.027	0.021	0.025	**0.042**
	NDCG@10	0.012	0.029	0.025	0.030	**0.063**
	NDCG@15	0.017	0.033	0.027	0.032	**0.072**

Table 5
Statistics of Runtime.

Dataset	GCN	GAT	APPNP	RGCN	HGT	HERE	PROFIT	TRMFG (ours)
Zhihu	6.01 s	6.46 s	6.03 s	6.17 s	6.65 s	7.49 s	6.77 s	14.22 s
Stack overflow	0.87 s	1.08 s	0.81 s	0.92 s	1.31 s	1.30 s	1.14 s	2.75 s
Zhuanzhi	0.29 s	0.33 s	0.28 s	0.30 s	0.39 s	0.35 s	0.36 s	1.28 s

Table 6
Statistics of parameters.

Model	GCN	GAT	APPNP	RGCN	HGT	HERE	PROFIT	TRMFG (ours)
Params	1.68M	1.69M	1.12M	3.38M	12.39M	66.60M	8.44M	14.66M

Table 7
Time complexity of GAT, HGT, and TRMFG.

Model	GAT	HGT	TRMFG (ours)
Complexity	$\mathbb{O}((q+t)d^2+ted)$	$\mathbb{O}((q+t)(d^2h+ed))$	$\mathbb{O}((pq+t)(d^2h+ed))$

5.6. Analysis of runtime and scalability

We conduct further investigations on the runtime and scalability of TRMFG to analyze the cost implications associated with our model.

Runtime We conduct experiments on the testing set to determine the average runtime of both baseline models and our model TRMFG, with the objective of evaluating the cost implications.

The results of the runtime in Table 5 provide the following insights:

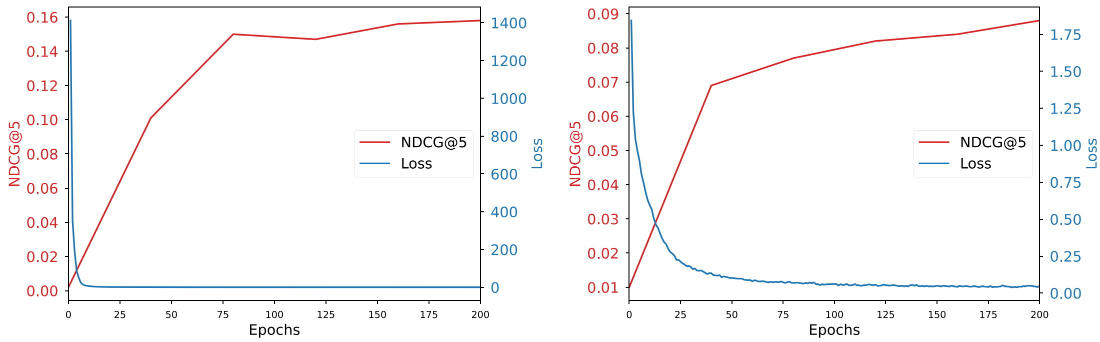
- During the time cost assessment, our TRMFG model demonstrates longer processing time compared to the baseline models across all three datasets. This can be attributed to the inherent complexity of our model, which incorporates multiple facets, while the baseline models feature one facet.
- The time required for testing is related to the number of tags. Our evaluation process requires the computation of scores for each tag in response to a given question and identifies the most optimal tags.

Scalability We also investigate the number of parameters of our model and baselines.

The analysis of the parameters presented in Table 6 provides the following insights:

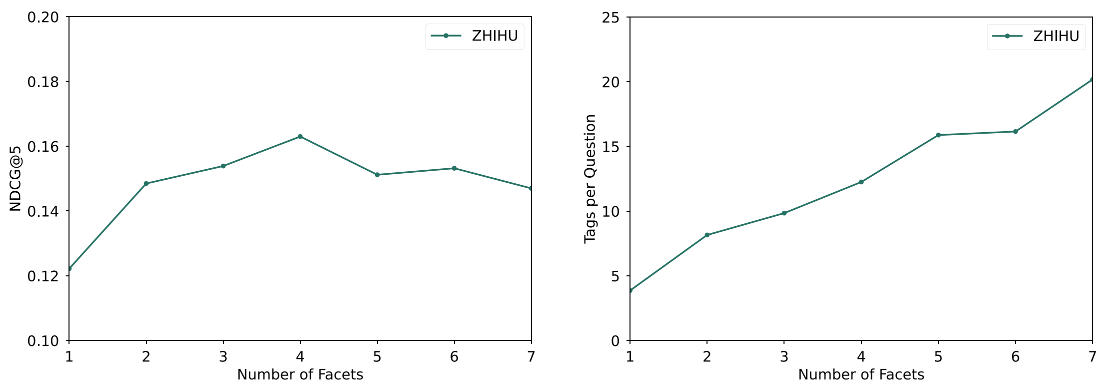
- Our model TRMFG exhibits a higher degree of complexity when compared to the GNN baselines, as it boasts a larger parameter count than the majority of these baseline models. However, it is important to note that the overall parameter count remains within reasonable limits.
- Compared with the state-of-the-art model HERE, our model features a reduced parameter count. This observation indicates that our model offers enhancements and is suitable for practical applications.

We analyze the time complexity of our TRMFG model. We assume that t is the number of tag nodes and q is the number of question nodes. The dimension of input and output vectors is d. The average number of neighbors is e and the number of attention heads is h. For our TRMFG model, we assume the number of facets is p. We analyze the time complexity of our TRMFG model compared to GAT and HGT, which are graph neural networks with attention mechanisms (see Table 7). Thus, the time complexity of TRMFG is on the same order of magnitude as the time complexity of the baselines, which is acceptable.



(a) Results on Zhihu dataset. (b) Results on Stack Overflow dataset.

Fig. 6. Loss vs. NDCG@5 of our TRMFG model.



(a) NDCG@5 (b) Retrieved tags per question

Fig. 7. Analysis of Multiple Facets.

5.7. Analysis of convergence

We investigate the convergence of our TRMFG model by evaluating the loss value and the NDCG@5 scores with increasing epochs.

As shown in Fig. 6(a), the loss of the Zhihu dataset decreases a lot in the first 10 epochs, followed by a steady decline until reaching convergence. Concurrently, the NDCG@5 demonstrates a consistent upward trend. The loss of the Stack Overflow dataset decreases a lot in the first 50 epoch while the NDCG@5 displays continuous enhancement. These results indicate that our TRMFG model exhibits a good convergence property.

5.8. Analysis of multiple facets

We conduct further investigations to assess the impact of varying the number of facets on our model. The results on the Zhihu dataset are presented in Fig. 7.

NDCG@5 We first analyze the quality of the predicted tag. It is worth noting that the model's performance of NDCG@5 improves significantly when utilizing four facets compared to other configurations. The results show a gradual improvement from one facet to four facets. However, beyond the four facets, there is a slight reduction in performance. These results suggest that the number of facets plays an important role in enhancing the model's performance, while four facets are adequate for the question tagging task. The application of our model in CQA websites, where relations can be complex and contain considerable noise, demonstrates the benefits of including multiple facets in the analysis.

Retrieved tags per question In addition to achieving accurate results, we also investigate the correlation between the number of facets and the number of retrieved tags for each question. To conduct this analysis, we employ a score threshold to filter the scores between tags and questions, focusing on the more relevant tags. It becomes evident that the TRMFG model with a higher number of facets exhibits a consistent trend of retrieving more tags for each question. This observation highlights the significance of including multiple facets in the TRMFG model, which leads to a more comprehensive understanding of the semantics of questions.

Consequently, the proposed model exhibits the capability to generate a wider array of relevant tags, leading to an improved overall retrieval performance. By appropriately increasing the number of facets, the questions can obtain more relevant and accurate tags, thereby positively impacting the question tagging task.

5.9. Case study of tri-relational question-tag graph

We select the results from our model TRMFG and the existing question tagging model HERE to demonstrate the necessity of relations in the question tagging task.

From the results in Fig. 8, we have the following observations:

- In the first example, the presence of the seen tag 'Movie' contributes to the accurate tagging of the question with 'Movie Commentary', 'Movie Recommendation'. With the semantic 'young' in the question, the true tag 'Youth Movie' is also retrieved for the question. Although other retrieved tags like 'Japanese drama' and 'Korean Drama' are not the ground truth, they are reasonable for users. However, the HERE model, which only focuses on tag relations from parent nodes to child nodes, produces inappropriate tags like 'Drama'. What is more, completely unrelated tag like 'Comprehensive and Progressive Trans Pacific Partnership Agreement' is retrieved by the HERE model. It may be attributed to other retrieved tags 'American Drama' and 'Korean Drama'.

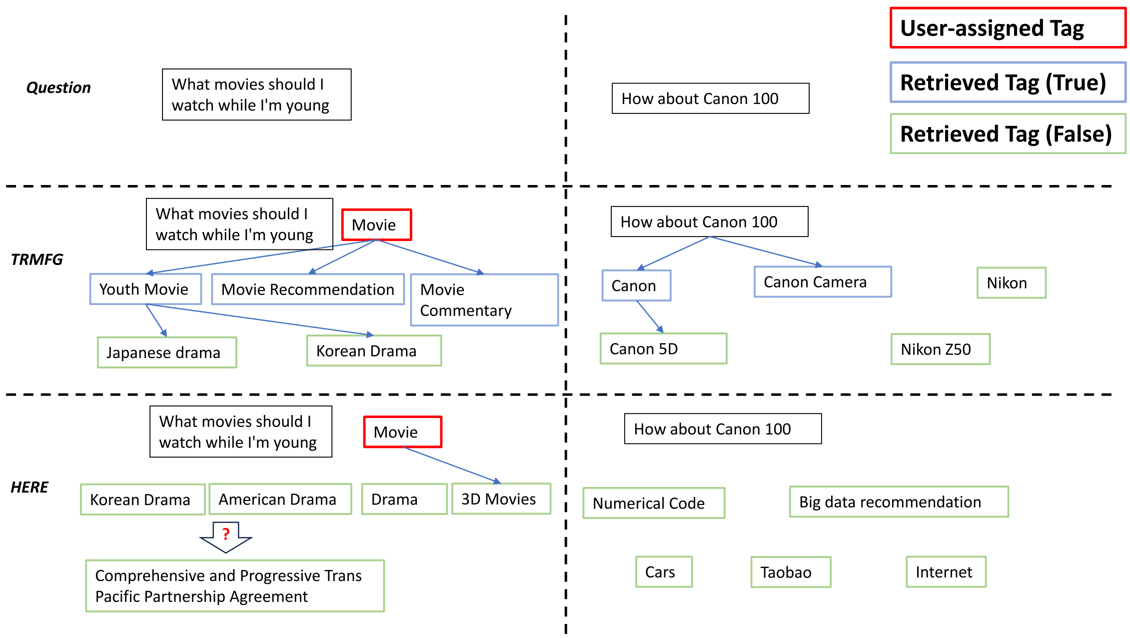


Fig. 8. Case Study of Tri-Relational Question-Tag Graph.

- In the second example, even though no seen tag is available, the semantic contexts of the question enable the TRMFG model to correctly identify and assign the tags 'Canon' and 'Canon Camera'. On the oppose, the HERE model also associates some parent nodes of the ground truth such as 'Numerical Code', to tag the question, which is not contextually relevant to the question.
- In both examples, the TRMFG model performs well with or without user-assigned tags. While the HERE model has possible interference from unimportant information.

With the tri-relational question-tag graph, our model can focus on both relations and semantic context to provide more accurate and contextually appropriate tags compared to the HERE model.

5.10. Qualitative results

We conduct a visualization analysis of our models and some baselines to demonstrate the effectiveness of our model. Specifically, we select three questions and their tags from the Zhihu dataset. To conduct a comparative evaluation, we include four baseline models alongside our TRMFG model. The selected baselines comprise the following:

1. **GCN** - chosen for its isomorphic graph network characteristics.
2. **APPNP** - due to its improved adjacent feature propagation capabilities.
3. **HGT** - as it is a heterogeneous graph neural network.
4. **HERE** - as it is an existing question-tagging model.

In Table 8, the experimental results of our model are more accurate, with the ground truth tags not only retrieved but have high priority, while other baselines cannot retrieve tags exactly. We have some observations of the qualitative results according to Table 8:

- For all the questions in the table, the isomorphic graph network GCN and APPNP fail to retrieve the ground truth tags. It indicates that focusing on original question semantics only is not enough for automatic question tagging.
- It is worth noting that in certain cases, such as Questions 1 and 2, the HGT model might perform as well as our TRMFG model because it also takes relations from tags to questions into consideration. However, it still struggles to focus on the most significant semantics. In Question 3, other tags may have higher priority to the ground truth in HGT model.
- The HERE model shows better results by being able to retrieve the ground truth tags, while these tags are mostly parent tags, as seen in Question 1 and Question 2. This could be attributed to the HERE model's focus on relations from parent tags to child tags.

Overall, the TRMFG demonstrates better performance, verifying the effectiveness of our proposed model.

5.11. Analysis of question routing

AQT is anticipated to augment the effectiveness of CQA-related tasks, such as question routing. To further investigate the effectiveness of our proposed model, we present visualization results of question routing on Zhihu dataset.

We have some observations of the qualitative results according to Fig. 9:

- The initial question 'Which delicacy in the novel do you always remember?' retrieves two tags: 'Novel' and 'Delicacy'.
- For users interested in the retrieved tag 'Novel', the CQA website recommends the question 'Which novel has a brilliant ending?', which retrieves another tag 'Literature'. Then another question 'Who is your favorite writer?' is recommended to users interested in tag 'Literature'.
- Similarly, for users interested in the retrieved tag 'Delicacy', the CQA website recommends the question 'What delicacy did you accidentally make?', which retrieves another tag 'Cooking'. Then another question 'What clever cooking techniques do you have?' will be recommended to users interested in tag 'Cooking'.

As illustrated in Fig. 9, we employ our model to retrieve relevant tags for the input questions. Subsequently, CQA websites can recommend questions related to the common tags to users.

5.12. Analysis of polysemy and homonymy

We have discussed the polysemy and homonymy problems in Section 5.1 Datasets. Specifically, we have tried to deal with the problem

Table 8
Visualization of the model results. The ground truth tags are bold if they appear in the model results.

Question	TRMFG	GCN	APPNP	HGT	HERE	Ground truth
What are some precautions when accepting a phone interview?	**Interview**	Junior college students	Contact	**Interview**	**Interview**	Interview
	Telephone	Information Technology (IT)	Head-hunting company	Interview Skills	Resume delivery	
	Consulting service	Postgraduate recommendation	Headhunting	Contact	Post	
	Website	Subject	Interview Skills	Internship	Contact	
	Language	Signal communication	Recruitment Skills	Workplace	Telephone	
What are some Podcast programs worth recommending?	**Podcast program**	Movies	Video Recom- mendations	**Podcast program**	**Podcast program**	Podcast program
	Talk Show	Information Technology (IT)	Broadcast production	Internet	Selection and Beauty (Podcast)	
	Study	Ying Ku	Radio program	Social networks	The Way of Taste (Podcast)	
	(Hi) story (Podcast)	Subject	MTV	Music variety shows	Popular communication (Podcast)	
	IOS Podcast app	Life, Art, Culture, and Activities	Radio Programs	Lifestyle	Crazy casting circle (Podcast)	
Which bookstores in Guangzhou are more distinctive?	**Bookstore**	Taobao online merchant	Beijing, Shanghai, Guangzhou	Life	Life	Bookstore
	Life	Double 11 Shopping Carnival	Beijing, Shanghai, Guangzhou, Shenzhen	**Bookstore**	Shenzhen City	
	Page Layout	Beijing	Provincial capital	Chinese cities	Life attitude	
	Read	Living abroad	Chain bookstore	Consumption	Guangzhou Life	
	Café	Buying Books	Bookstore recommendation	Book	Shenzhen Life	

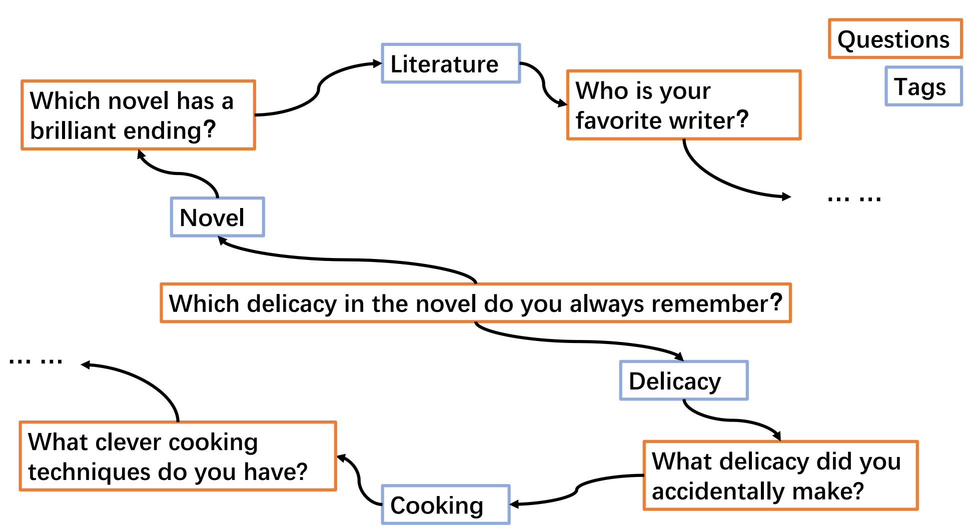


Fig. 9. Example of question routing. For the original question, AQT models retrieve tags to describe questions. Subsequently, questions assigned with the same tags are recommended to users in CQA websites.

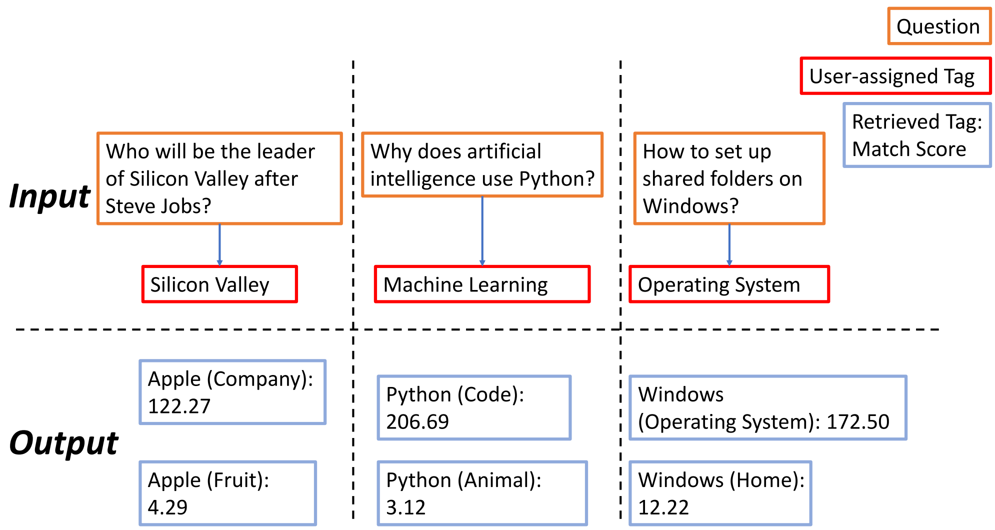


Fig. 10. Example of dealing with polysemy problem.

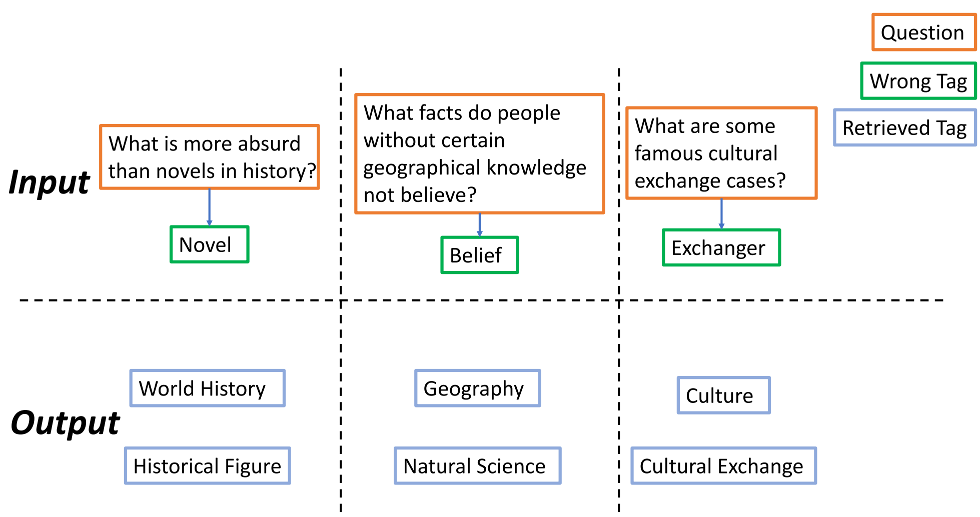


Fig. 11. Example of correcting closed questions.

of homonymy through the utilization of RoBERTa model. Moreover, to investigate the effectiveness of our TRMFG model in handling the polysemy problem of tags, we analyze the results on Zhihu dataset.

We have some observations of the qualitative results according to Fig. 10:

- In the first example, the words 'Steve Jobs' and the user-assigned tag 'Silicon Valley' exhibit associations with Apple Company. Therefore, the expected tag 'Apple (company)' receives a significant score of 122.27, surpassing that of the tag 'Apple (fruit)'.
- In the second example, the phrase 'artificial intelligence' in the question and the user-assigned tag 'Machine Learning' enable the model to learn that the question focuses on data coding. Then it assigns a higher score to the tag 'Python (Code)'.
- In the third example, the user-assigned tag 'Operating System' aids the model in comprehending the question's semantics, leading to the retrieval of the tag 'Windows (Operating System)'.

Therefore, our model demonstrates a capacity to distinguish the polysemy of tags. The experimental results that tags with accurate meanings receive higher scores from our TRMFG model emphasize its capability to address the challenges of polysemy and homonymy within the domain of AQT.

5.13. Analysis of closed questions

In CQA websites, some questions may have incorrect assigned tags and be closed due to the lack of visibility. Thus, we select examples from results on the Zhihu dataset to validate the ability of our TRMFG model to correct such instances by retrieving proper tags for questions that initially have incorrect user-assigned tags.

We have some observations of the qualitative results according to Fig. 11:

- In the first example, the wrong tag 'Novel' misinterprets the semantics of the question. On the oppose, our model retrieves the proper tags 'World History' and 'Historical Figure', which accurately describe the question.
- In the second example, the question centers around geographical knowledge, and the tags 'Geography' and 'Natural Science' retrieved by our model are precise.

• In the third example, the wrong tag focuses solely on the word ‘exchange’ in the question. However, our model TRMFG can learn the context of the question and retrieve the correct tags ‘Culture’ and ‘Cultural Exchange’ for the question.

These experimental results indicate that our model can incorporate tag-question relations and retrieve relevant tags for questions, even in cases where the questions initially have incorrect user-assigned tags.

6. Limitation and threats to validity

Although our TRMFG approach models the questions, tags, and relations as well as learns informative node features for question tagging in CQA websites, it still has some limitations: (1) We do not consider the concept of continual learning within the TRMFG framework. In instances where a substantial volume of new tags is introduced into the database, it may undertake the retraining of the model to update the features associated with tag nodes. (2) Furthermore, our model does not account for open-domain challenges. In situations where certain CQA websites are constrained in users’ ability to freely create new tags, we have conducted experiments focusing exclusively on existing tags within the dataset we have collected. We will try to work on open-domain challenges to deal with missing tag information in future work.

7. Conclusion

In this paper, we propose the Tri-Relational Multi-Faceted Graph Neural Networks for Question Tagging (TRMFG) approach, facilitating the automatic question tagging task in CQA websites. We design the Tri-Relational Question-Tag Graph to model the questions and tags in CQA websites. Also, we propose Tri-Relational Question-Tag GNN to handle diverse types of relations between questions and tags, allowing for the capture of hidden latent representation from complex relations. Specially, we design Multi-Faceted Question GNN to capture semantics from user-assigned tags for questions. By encoding questions into multiple facets of vectors, our model provides questions with a higher likelihood of extracting semantics from user-assigned tags, leading to improved performance in the automatic question tagging task. Through extensive experiments, we demonstrate that our TRMFG model can automatically tag questions in CQA websites with improved accuracy, validating the effectiveness and utility of our proposed approach in addressing the automatic question tagging challenge in CQA websites.

CRediT authorship contribution statement

Nuojia Xu: Conceptualization, Methodology, Investigation, Writing – original draft, Data curation, Visualization. **Jun Hu:** Conceptualization, Writing – review & editing, Data curation. **Quan Fang:** Conceptualization, Supervision, Writing – review & editing. **Dizhan Xue:** Visualization, Writing – review & editing. **Yongxi Li:** Investigation, Writing – review & editing. **Shengsheng Qian:** Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work is supported in part by the National Natural Science Foundation of China under Grants 62036012, 62072456, 62106262. This work is being sponsored by SMP-IDATA Open Youth Fund. This work is also supported by the Beijing Natural Science Foundation, China (L221004, JQ23018).

References

[1] Y. Deng, W. Lam, Y. Xie, D. Chen, Y. Li, M. Yang, Y. Shen, Joint learning of answer selection and answer summary generation in community question answering, in: 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, February 7-12, 2020, AAAI Press, 2020, pp. 7651–7658, URL https://ojs.aaai.org/index.php/AAAI/article/view/6266.

[2] Y. Xie, Y. Shen, Y. Li, M. Yang, K. Lei, Attentive user-engaged adversarial neural network for community question answering, in: 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, February 7-12, 2020, AAAI Press, 2020, pp. 9322–9329, URL https://ojs.aaai.org/index.php/AAAI/article/view/6472.

[3] J. Fu, Y. Li, Q. Zhang, Q. Wu, R. Ma, X. Huang, Y. Jiang, Recurrent memory reasoning network for expert finding in community question answering, in: J. Caverlee, X.B. Hu, M. Lalmas, W. Wang (Eds.), WSDM ’20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020, ACM, 2020, pp. 187–195, http://dx.doi.org/10.1145/3336191.3371817.

[4] J. Hu, S. Qian, Q. Fang, C. Xu, Hierarchical graph semantic pooling network for multi-modal community question answer matching, in: L. Amsaleg, B. Huet, M.A. Larson, G. Gravier, H. Hung, C. Ngo, W.T. Ooi (Eds.), Proceedings of the 27th ACM International Conference on Multimedia, MM 2019, Nice, France, October 21-25, 2019, ACM, 2019, pp. 1157–1165, http://dx.doi.org/10.1145/3343031.3350966.

[5] X. Wang, C. Huang, L. Yao, B. Benatallah, M. Dong, A survey on expert recommendation in community question answering, J. Comput. Sci. Tech. 33 (4) (2018) 625–653, http://dx.doi.org/10.1007/s11390-018-1845-0.

[6] Y. Chen, A. Subburathinam, C. Chen, M.J. Zaki, Personalized food recommendation as constrained question answering over a large-scale food knowledge graph, in: L. Lewin-Eytan, D. Carmel, E. Yom-Tov, E. Agichtein, E. Gabrilovich (Eds.), WSDM ’21, the Fourteenth ACM International Conference on Web Search and Data Mining, Virtual Event, Israel, March 8-12, 2021, ACM, 2021, pp. 544–552, http://dx.doi.org/10.1145/3437963.3441816.

[7] M. Zhao, A.L. Jia, A dual-attention heterogeneous graph neural network for expert recommendation in online agricultural question and answering communities, in: 25th IEEE International Conference on Computer Supported Cooperative Work in Design, CSCWD 2022, Hangzhou, China, May 4-6, 2022, IEEE, 2022, pp. 926–931, http://dx.doi.org/10.1109/CSCWD54268.2022.9776176.

[8] N. Ghasemi, R. Fatourechi, S. Momtazi, User embedding for expert finding in community question answering, ACM Trans. Knowl. Discov. Data 15 (4) (2021) 70:1–70:16, http://dx.doi.org/10.1145/3441302.

[9] A. Askari, S. Verberne, G. Pasi, Expert finding in legal community question answering, in: M. Hagen, S. Verberne, C. Macdonald, C. Seifert, K. Balog, K. Nørvåg, V. Setty (Eds.), Advances in Information Retrieval - 44th European Conference on IR Research, ECIR 2022, Stavanger, Norway, April 10-14, 2022, Proceedings, Part II, in: Lecture Notes in Computer Science, vol. 13186, Springer, 2022, pp. 22–30, http://dx.doi.org/10.1007/978-3-030-99739-7_3.

[10] M. Dehghan, A.A. Abin, M. Neshati, An improvement in the quality of expert finding in community question answering networks, Decis. Support Syst. 139 (2020) 113425, http://dx.doi.org/10.1016/j.dss.2020.113425.

[11] H. Wu, W. Wu, M. Zhou, E. Chen, L. Duan, H. Shum, Improving search relevance for short queries in community question answering, in: B. Carterette, F. Diaz, C. Castillo, D. Metzler (Eds.), Seventh ACM International Conference on Web Search and Data Mining, WSDM 2014, New York, NY, USA, February 24-28, 2014, ACM, 2014, pp. 43–52, http://dx.doi.org/10.1145/2556195.2556239.

[12] Y. Qin, Z. Cai, D. Jin, L. Yan, S. Liang, K. Zhu, Y. Lin, X. Han, N. Ding, H. Wang, R. Xie, F. Qi, Z. Liu, M. Sun, J. Zhou, WebCPM: Interactive web search for Chinese long-form question answering, in: A. Rogers, J.L. Boyd-Graber, N. Okazaki (Eds.), Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023, Association for Computational Linguistics, 2023, pp. 8968–8988, http://dx.doi.org/10.18653/v1/2023.acl-long.499.

[13] J. Huang, D. Tang, L. Shou, M. Gong, K. Xu, D. Jiang, M. Zhou, N. Duan, Cosqa: 20, 000+ web queries for code search and question answering, in: C. Zong, F. Xia, W. Li, R. Navigli (Eds.), Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021, Association for Computational Linguistics, 2021, pp. 5690–5700, http://dx.doi.org/10.18653/v1/2021.acl-long.442.

[14] V. V., M.K. Mohania, V. Goyal, TagRec: Automated tagging of questions with hierarchical learning taxonomy, in: Y. Dong, N. Kourtellis, B. Hammer, J.A. Lozano (Eds.), Machine Learning and Knowledge Discovery in Databases. Applied Data Science Track - European Conference, ECML PKDD 2021, Bilbao, Spain, September 13-17, 2021, Proceedings, Part V, in: Lecture Notes in Computer Science, vol. 12979, Springer, 2021, pp. 381–396, http://dx.doi.org/10.1007/978-3-030-86517-7_24.

[15] Y. Xiang, H. Wang, D. Ji, Z. Zhang, J. Zhu, Neutag's classification system for zhihu questions tagging task, in: M. Zhang, V. Ng, D. Zhao, S. Li, H. Zan (Eds.), Natural Language Processing and Chinese Computing - 7th CCF International Conference, NLPCC 2018, Hohhot, China, August 26-30, 2018, Proceedings, Part I, in: Lecture Notes in Computer Science, vol. 11108, Springer, 2018, pp. 279–288, http://dx.doi.org/10.1007/978-3-319-99495-6_24.

[16] C. Tian, Y. Xie, Y. Li, N. Yang, W.X. Zhao, Learning to denoise unreliable interactions for graph collaborative filtering, in: E. Amigó, P. Castells, J. Gonzalo, B. Carterette, J.S. Culpepper, G. Kazai (Eds.), SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022, ACM, 2022, pp. 122–132, http://dx.doi.org/10.1145/3477495.3531889.

[17] Z. Hu, Y. Dong, K. Wang, Y. Sun, Heterogeneous graph transformer, in: Y. Huang, I. King, T. Liu, M. van Steen (Eds.), WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020, ACM / IW3C2, 2020, pp. 2704–2710, http://dx.doi.org/10.1145/3366423.3380027.

[18] L. Nie, Y. Li, F. Feng, X. Song, M. Wang, Y. Wang, Large-scale question tagging via joint question-topic embedding learning, ACM Trans. Inf. Syst. 38 (2) (2020) 20:1–20:23, http://dx.doi.org/10.1145/3380954.

[19] X. Zhang, M. Liu, J. Yin, Z. Ren, L. Nie, Question tagging via graph-guided ranking, ACM Trans. Inf. Syst. 40 (1) (2022) 12:1–12:23, http://dx.doi.org/10.1145/3468270.

[20] A. Tanwar, D.K. Vishwakarma, A deep neural network-based hybrid recommender system with user-user networks, Multim. Tools Appl. 82 (10) (2023) 15613–15633, http://dx.doi.org/10.1007/S11042-022-13936-3.

[21] E. Lima, W. Shi, X. Liu, Q. Yu, Integrating multi-level tag recommendation with external knowledge bases for automatic question answering, ACM Trans. Internet Technol. 19 (3) (2019) 34:1–34:22, http://dx.doi.org/10.1145/3319528.

[22] M. Ahmadian, S. Ahmadian, M. Ahmadi, RDERL: reliable deep ensemble reinforcement learning-based recommender system, Knowl.-Based Syst. 263 (2023) 110289, http://dx.doi.org/10.1016/J.KNOSYS.2023.110289.

[23] D. Haraguchi, M. Otani, Coarse-to-fine font recommendation for banner designs, in: Companion Proceedings of the 28th International Conference on Intelligent User Interfaces, IUI 2023, Sydney, NSW, Australia, March 27-31, 2023, ACM, 2023, pp. 148–150, http://dx.doi.org/10.1145/3581754.3584156.

[24] M. Izadi, M. Nejati, A. Heydarnoori, Semantically-enhanced topic recommendation systems for software projects, Empir. Softw. Eng. 28 (2) (2023) 50, http://dx.doi.org/10.1007/S10664-022-10272-W.

[25] W. Yoon, R. Jackson, A. Lagerberg, J. Kang, Sequence tagging for biomedical extractive question answering, Bioinform. 38 (15) (2022) 3794–3801, http://dx.doi.org/10.1093/bioinformatics/btac397.

[26] J. Song, X. Xu, X. Wang, TSAR-based expert recommendation mechanism for community question answering, in: W. Shen, J.A. Barthès, J. Luo, Y. Shi, J. Zhang (Eds.), 24th IEEE International Conference on Computer Supported Cooperative Work in Design, CSCWD 2021, Dalian, China, May 5-7, 2021, IEEE, 2021, pp. 162–167, http://dx.doi.org/10.1109/CSCWD49262.2021.9437843.

[27] J. Gomes Jr., R.C. de Mello, V. Ströele, J.F. de Souza, A hereditary attentive template-based approach for complex knowledge base question answering systems, Expert Syst. Appl. 205 (2022) 117725, http://dx.doi.org/10.1016/J.ESWA.2022.117725.

[28] L.M. de Campos, J.M. Fernández-Luna, J.F. Huete, Use of topical and temporal profiles and their hybridisation for content-based recommendation, User Model. User Adapt. Interact. 33 (4) (2023) 911–937, http://dx.doi.org/10.1007/S11257-022-09354-7.

[29] G. Costa, R. Ortale, *Here are the answers. What is your question?* Bayesian collaborative tag-based recommendation of time-sensitive expertise in question-answering communities, Expert Syst. Appl. 225 (2023) 120042, http://dx.doi.org/10.1016/J.ESWA.2023.120042.

[30] G. Costa, R. Ortale, Collaborative recommendation of temporally-discounted tag-based expertise for community question answering, in: H.W. Lauw, R.C. Wong, A. Ntoulas, E. Lim, S. Ng, S.J. Pan (Eds.), Advances in Knowledge Discovery and Data Mining - 24th Pacific-Asia Conference, PAKDD 2020, Singapore, May 11-14, 2020, Proceedings, Part I, in: Lecture Notes in Computer Science, vol. 12084, Springer, 2020, pp. 41–52, http://dx.doi.org/10.1007/978-3-030-47426-3_4.

[31] A.M. Rocha, M. de Almeida Maia, Mining relevant solutions for programming tasks from search engine results, IET Softw. 17 (4) (2023) 455–471, http://dx.doi.org/10.1049/SFW2.12127.

[32] M. Tsai, Y. Wu, User intent prediction search engine system based on query analysis and image recognition technologies, J. Supercomput. 79 (5) (2023) 5327–5359, http://dx.doi.org/10.1007/S11227-022-04874-W.

[33] F. Wu, A.H.S. Jr., T. Zhang, C. Fifty, T. Yu, K.Q. Weinberger, Simplifying graph convolutional networks, in: K. Chaudhuri, R. Salakhutdinov (Eds.), Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA, in: Proceedings of Machine Learning Research, 97, PMLR, 2019, pp. 6861–6871, URL http://proceedings.mlr.press/v97/wu19e.html.

[34] W. Wang, W. Zhang, S. Liu, Q. Liu, B. Zhang, L. Lin, H. Zha, Beyond clicks: Modeling multi-relational item graph for session-based target behavior prediction, in: Y. Huang, I. King, T. Liu, M. van Steen (Eds.), WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020, ACM / IW3C2, 2020, pp. 3056–3062, http://dx.doi.org/10.1145/3366423.3380077.

[35] X. Wang, X. He, Y. Cao, M. Liu, T. Chua, KGAT: knowledge graph attention network for recommendation, in: A. Teredesai, V. Kumar, Y. Li, R. Rosales, E. Terzi, G. Karypis (Eds.), Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019, ACM, 2019, pp. 950–958, http://dx.doi.org/10.1145/3292500.3330989.

[36] S. Qian, D. Xue, H. Zhang, Q. Fang, C. Xu, Dual adversarial graph neural networks for multi-label cross-modal retrieval, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, No. 3, 2021, pp. 2440–2448.

[37] S. Qian, D. Xue, Q. Fang, C. Xu, Adaptive label-aware graph convolutional networks for cross-modal retrieval, IEEE Trans. Multimed. 24 (2021) 3520–3532.

[38] S. Qian, D. Xue, Q. Fang, C. Xu, Integrating multi-label contrastive learning with dual adversarial graph neural networks for cross-modal retrieval, IEEE Trans. Pattern Anal. Mach. Intell. 45 (4) (2022) 4794–4811.

[39] J. Hu, S. Qian, Q. Fang, Y. Wang, Q. Zhao, H. Zhang, C. Xu, Efficient graph deep learning in TensorFlow with tf_geometric, in: H.T. Shen, Y. Zhuang, J.R. Smith, Y. Yang, P. César, F. Metze, B. Prabhakaran (Eds.), MM '21: ACM Multimedia Conference, Virtual Event, China, October 20 - 24, 2021, ACM, 2021, pp. 3775–3778, http://dx.doi.org/10.1145/3474085.3478322.

[40] S. Qian, H. Chen, D. Xue, Q. Fang, C. Xu, Open-world social event classification, in: Proceedings of the ACM Web Conference 2023, 2023, pp. 1562–1571.

[41] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, 2016, CoRR abs/1609.02907.URL http://arxiv.org/abs/1609.02907.

[42] M. Defferrard, X. Bresson, P. Vandergheynst, Convolutional neural networks on graphs with fast localized spectral filtering, in: D.D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, R. Garnett (Eds.), Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain, 2016, pp. 3837–3845, URL https://proceedings.neurips.cc/paper/2016/hash/04df4d434d481c5bb723be1b6df1ee65-Abstract.html.

[43] W.L. Hamilton, Z. Ying, J. Leskovec, Inductive representation learning on large graphs, in: I. Guyon, U. von Luxburg, S. Bengio, H.M. Wallach, R. Fergus, S.V.N. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, 2017, pp. 1024–1034, URL https://proceedings.neurips.cc/paper/2017/hash/5dd9db5e033da9c6fb5ba83c7a7ebea9-Abstract.html.

[44] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, in: Y. Bengio, Y. LeCun (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015, URL http://arxiv.org/abs/1409.0473.

[45] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, Y. Bengio, Graph attention networks, 2017, CoRR abs/1710.10903. URL http://arxiv.org/abs/1710.10903.

[46] S. Brody, U. Alon, E. Yahav, How attentive are graph attention networks? in: The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022, OpenReview.net, 2022, URL https://openreview.net/forum?id=F72ximsx7C1.

[47] H. Chen, H. Yin, W. Wang, H. Wang, Q.V.H. Nguyen, X. Li, PME: projected metric embedding on heterogeneous networks for link prediction, in: Y. Guo, F. Farooq (Eds.), Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018, ACM, 2018, pp. 1177–1186, http://dx.doi.org/10.1145/3219819.3219986.

[48] B. Hu, Y. Fang, C. Shi, Adversarial learning on heterogeneous information networks, in: A. Teredesai, V. Kumar, Y. Li, R. Rosales, E. Terzi, G. Karypis (Eds.), Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019, ACM, 2019, pp. 120–129, http://dx.doi.org/10.1145/3292500.3330970.

[49] X. Wang, X. He, M. Wang, F. Feng, T. Chua, Neural graph collaborative filtering, in: B. Piwowarski, M. Chevalier, E. Gaussier, Y. Maarek, J. Nie, F. Scholer (Eds.), Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019, ACM, 2019, pp. 165–174, http://dx.doi.org/10.1145/3331184.3331267.

[50] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, M. Wang, LightGCN: Simplifying and powering graph convolution network for recommendation, in: J.X. Huang, Y. Chang, X. Cheng, J. Kamps, V. Murdock, J. Wen, Y. Liu (Eds.), Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020, ACM, 2020, pp. 639–648, http://dx.doi.org/10.1145/3397271.3401063.

[51] Y. Su, R. Zhang, S.M. Erfani, J. Gan, Neural graph matching based collaborative filtering, in: F. Diaz, C. Shah, T. Suel, P. Castells, R. Jones, T. Sakai (Eds.), SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021, ACM, 2021, pp. 849–858, http://dx.doi.org/10.1145/3404835.3462833.

[52] K. Li, C. Wang, J. Lai, H. Yuan, Self-supervised group graph collaborative filtering for group recommendation, in: T. Chua, H.W. Lauw, L. Si, E. Terzi, P. Tsaparas (Eds.), Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, WSDM 2023, Singapore, 27 February 2023 - 3 March 2023, ACM, 2023, pp. 69–77, http://dx.doi.org/10.1145/3539597.3570400.

[53] K. Gong, X. Song, S. Wang, S. Liu, Y. Li, ITSM-GCN: informative training sample mining for graph convolutional network-based collaborative filtering, in: M.A. Hasan, L. Xiong (Eds.), Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022, ACM, 2022, pp. 614–623, http://dx.doi.org/10.1145/3511808.3557368.

[54] J. Hu, S. Qian, Q. Fang, C. Xu, MGDCF: distance learning via Markov graph diffusion for neural collaborative filtering, 2022, http://dx.doi.org/10.48550/arXiv.2204.02338, CoRR abs/2204.02338. arXiv:2204.02338.

[55] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, 2018, CoRR abs/1810.04805. URL http://arxiv.org/abs/1810.04805.

[56] P. Prakash, J. Killingback, H. Zamani, Learning robust dense retrieval models from incomplete relevance labels, in: F. Diaz, C. Shah, T. Suel, P. Castells, R. Jones, T. Sakai (Eds.), SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021, ACM, 2021, pp. 1728–1732, http://dx.doi.org/10.1145/3404835.3463106.

[57] J. Klicpera, A. Bojchevski, S. Günnemann, Predict then propagate: graph neural networks meet personalized pagerank in: The Seventh International Conference on Learning Representations, ICLR 2019, Virtual Event, May 6-9, 2019, OpenReview.net, 2019, URL https://openreview.net/forum?id=H1gL-2A9Ym.

[58] M.S. Schlichtkrull, T.N. Kipf, P. Bloem, R. van den Berg, I. Titov, M. Welling, Modeling relational data with graph convolutional networks, 2017, CoRR abs/1703.06103. URL http://arxiv.org/abs/1703.06103.

[59] J. Demsar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (2006) 1–30, URL http://jmlr.org/papers/v7/demsar06a.html.

Nuojia Xu received the B.E. degree in computer science and technology from University of Chinese Academy of Sciences, Beijing, China, in 2021. She is currently a Master student at Institute of Automation, Chinese Academy of Sciences, Beijing, China. Her research interests are in the field of graph deep learning, and data mining.

Jun Hu received the Ph.D. degree from the School of Computer Science and Information Engineering, Hefei University of Technology, China, in 2020. He is currently a Research Fellow in the School of Computing in National University of Singapore. His research interests include graph deep learning and social multimedia.

Quan Fang is a Research Professor in School of Artificial Intelligence at Beijing University of Posts and Telecommunications. He received the B.E. degree from Beihang University, Beijing, China, in 2010 and received the Ph.D. degree from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2015.

He was awarded the 2013 Microsoft Research Asia Fellowship. He received the Best Student Paper in Internet Multimedia Modeling 2013 and the Best Paper Finalist in ACM Multimedia 2013/2019. He was awarded the Distinguished Doctoral Dissertation of Chinese Academy of Sciences and China Association for Artificial Intelligence. His research interest lies in Multimedia Knowledge Computing.

Dizhan Xue received the B.E. degree in computer science and technology from University of Chinese Academy of Sciences, Beijing, China, in 2021. He is currently a Ph.D student at Institute of Automation, Chinese Academy of Sciences, Beijing, China. His research interests are in the field of machine learning, cross-modal reasoning, and multimedia content analysis.

Yongxi Li received his Master degree from Beihang University in 2017. He then became an associate engineer in 2017 and is currently an associate researcher in Multimedia Computing Group (MMC), National Laboratory of Pattern Recognition, Institute of Automation Chinese Academy of Sciences. His research interests include noisy annotation and person re-identification.

Shengsheng Qian (Member, IEEE) received the B.E. degree from the Jilin University, Changchun, China, in 2012, and the Ph.D. degree in pattern recognition and intelligent systems from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2017. He is currently an Associate Professor with the Institute of Automation, Chinese Academy of Sciences. His current research interests include data mining and multimedia content analysis.