Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Heterogeneous Relational Graph Neural Networks with Adaptive Objective for End-to-End Task-Oriented Dialogue

Qingbin Liu [a,b], Guirong Bai [a,b], Shizhu He [a,b], Cao Liu [c], Kang Liu [a,b], Jun Zhao [a,b,*]

[a] *National Laboratory of Pattern Recognition, 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] *Meituan, Beijing, 100102, China*

ARTICLE INFO

Article history:
Received 19 January 2021
Received in revised form 12 April 2021
Accepted 29 May 2021
Available online 1 June 2021

Keywords:
End-to-end task-oriented dialogue
Heterogeneous relational graph neural networks
Shared-private parameterization
Hierarchical attention mechanism
Adaptive objective

ABSTRACT

End-to-end task-oriented dialogue systems, which provide a natural and informative way for human–computer interaction, are gaining more and more attention. The main challenge of such dialogue systems is how to effectively incorporate external knowledge bases into the learning framework. However, existing approaches usually overlook the natural graph structure information in the knowledge base and the relevant information between the knowledge base and the dialogue history, which makes them deficient in handling the above challenge. Besides, existing methods ignore the entity imbalance problem and treat different entities in system responses indiscriminately, which limits the learning of hard target entities. To address the two challenges, we propose Heterogeneous Relational Graph Neural Networks with Adaptive Objective (**HRGNN-AO**) for end-to-end task-oriented dialogue systems. In the method, we explore effective heterogeneous relational graphs to jointly capture multi-perspective graph structure information from the knowledge base and the dialogue history, which ultimately facilitates the generation of informative responses. Moreover, we design two components, shared-private parameterization and hierarchical attention mechanism, to solve the overfitting and confusion problems in the heterogeneous relational graph, respectively. To handle the entity imbalance problem, we propose an adaptive objective, which dynamically adjusts the weights of different target entities during the training process. The experimental results show that HRGNN-AO is effective in generating informative responses and outperforms state-of-the-art dialogue systems on the SMD and extended Multi-WOZ 2.1 datasets.

1. Introduction

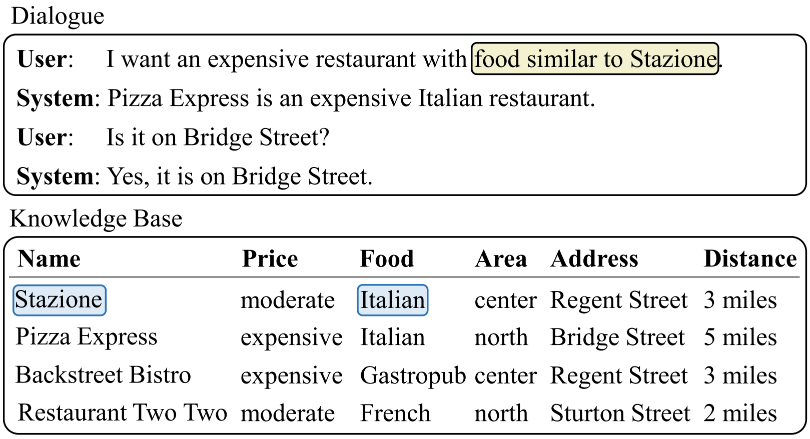
Task-oriented dialogue systems are widely used to help users accomplish a variety of tasks, such as booking restaurants, finding flights, and querying weather [1–5]. Since these dialogue systems provide a natural and informative way for human–machine interaction, they are attracting more and more attention from both industry and academia. Traditional pipeline solutions consist of natural language understanding, dialogue management, and natural language generation [6–8], where each module is designed separately. In order to reduce the human effort required to design and maintain these modules, recent work usually adopts an end-to-end approach to incorporate large-scale knowledge bases (KBs) into the learning framework and directly output system responses without separate modules [9–11].

The main challenge of end-to-end task-oriented dialogue systems is how to effectively incorporate external KBs into the learning framework [5,12]. Nevertheless, existing work [5,12,13] usually uses disordered memory to represent the knowledge base (KB), which ignores the natural graph structure information in the KB. As a result, existing work fails to capture sufficient information from the knowledge base to represent entities. In addition, during a dialogue, users usually focus on different relations. Take the dialogue in Fig. 1 as an example. In the first turn, the user focuses on two relations, “food” and “price”. Later, in the second turn, the user focuses on the “address” relation. Therefore, under different dialogue contexts, task-oriented dialogue systems should have the ability to dynamically capture graph structure information from the knowledge base to represent entities.

In fact, the knowledge base is also a powerful source of information for dialogue understanding. As shown in Fig. 1, in the first dialogue turn, the user expresses two preferences for the desired restaurant, “expensive” and “food similar to Stazione”. By understanding the dialogue history without accessing the knowledge base, the model is able to understand the first preference “expensive”, while it fails to understand the second preference “food similar to Stazione”. The key reason is that understanding

[*] Corresponding author at: National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China.
E-mail address: jzhao@nlpr.ia.ac.cn (J. Zhao).

https://doi.org/10.1016/j.knosys.2021.107186
0950-7051/© 2021 Elsevier B.V. All rights reserved.



Dialogue

User: I want an expensive restaurant with food similar to Stazione.

System: Pizza Express is an expensive Italian restaurant.

User: Is it on Bridge Street?

System: Yes, it is on Bridge Street.

Knowledge Base

Name	**Price**	**Food**	**Area**	**Address**	**Distance**
Stazione	moderate	Italian	center	Regent Street	3 miles
Pizza Express	expensive	Italian	north	Bridge Street	5 miles
Backstreet Bistro	expensive	Gastropub	center	Regent Street	3 miles
Restaurant Two Two	moderate	French	north	Sturton Street	2 miles

Fig. 1. Example of a task-oriented dialogue that incorporates a knowledge base. The yellow box represents a preference informed by the user. The blue boxes represent the knowledge related to the preference.

the second preference requires the model to provide and learn a relational path from the knowledge base to the dialogue history to obtain the "food" attribute of the restaurant "Stazione". Without the relational path, these dialogue systems tend to predict an incorrect restaurant, e.g., "Backstreet Bistro", which matches only part of the preferences, i.e., "expensive".

Besides, existing approaches usually ignore the entity imbalance problem and treat different entities in system responses indiscriminately. In detail, the frequency of different kinds of entities in the target system response is different. For example, in the extended Multi-WOZ 2.1 dataset [5,14,15], 35.7% responses contain entities belonging to the relation "name", while 4.0% for the relation "postcode". We refer to this problem as "entity imbalance", which varies the learning difficulty of different entities and limits the learning of hard entities.

To address these two issues, we propose Heterogeneous Relational Graph Neural Networks with Adaptive Objective (**HRGNN-AO**) for end-to-end task-oriented dialogue systems. In the method, we explore effective heterogeneous relational graphs that jointly encode the knowledge base and the dialogue history to capture the graph structure information. The constructed graph uses multiple relations to link two kinds of nodes, entities in the knowledge base and words in the dialogue history. As a result, the constructed graph has the following advantages: (1) It dynamically captures the graph structure information from the knowledge base according to the dialogue process to represent the entities. In contrast to ours, previous work [5,12] usually uses an embedding layer to represent the entities, which remains unchanged regardless of the dialogue process. (2) It propagates relevant information from the knowledge base to the dialogue context through relational paths, enabling knowledge-aware dialogue understanding. (3) The constructed graph is scalable in different domains. For example, we can bring nodes from different domains into the graph for multi-domain dialogues. However, since the heterogeneous relational graph contains a large number of imbalanced relations, it is suboptimal to directly deploy existing Relational Graph Neural Networks (RGNNs) [16–19] on the constructed graph. It generally suffers from two problems, overfitting and confusion. **Overfitting**: The relations in the heterogeneous relational graph are imbalanced, which makes it easy for the model to overfit those rare relations [16,17]. **Confusion**: When aggregating relational information, existing RGNNs usually treat all relations as equally important. Therefore, they are difficult to capture important relational information from a large number of relations [16–18]. To handle the two problems, we design two components: a shared-private parameterization and a hierarchical attention mechanism. To address the overfitting problem, we propose a shared-private parameterization module, which transfers effective features from high-frequency relations to rare relations. To reduce the adverse effects of confusion, we design a hierarchical attention mechanism to make the model pay more attention to important relations. Furthermore, for the entity imbalance problem, we propose an adaptive objective to dynamically evaluate the difficulty in a performance-sensitive manner. Then, we adaptively adjust the learning weights of different target entities to balance the learning process as much as possible.

The experimental results on the two multi-domain task-oriented dialogue datasets, SMD [9] and extended Multi-WOZ 2.1 [5,14,15], demonstrate that our method outperforms previous state-of-the-art methods by 2.5% and 2.9% in terms of the entity F1 metric, respectively. In addition, the ablation experiments further demonstrate the effectiveness of our proposed method.

The main contributions of this paper are as follows:

- In the end-to-end task-oriented dialogue task, we propose heterogeneous relational graph neural networks that jointly encode the knowledge base and the dialogue history to capture the graph structure information, which ultimately facilitates the generation of informative responses.
- To handle the overfitting and confusion problems in the constructed heterogeneous relational graph, we design two modules, the shared-private parameterization and the hierarchical attention mechanism.
- We propose an adaptive objective to solve the entity imbalance problem by dynamically adjusting the weights of different kinds of entities. To the best of our knowledge, our method is the first to address the entity imbalance problem in task-oriented dialogues.
- Experimental results show that our method outperforms previous state-of-the-art methods. In addition, the ablation experiments further demonstrate the effectiveness of our proposed method.

2. Related work

2.1. Task-oriented dialogue system

Task-oriented dialogue is a hot research topic in recent years, where typical task-oriented dialogue systems can be divided into two categories: pipeline solutions and end-to-end dialogue models. The pipeline solution consists of three modules: natural language understanding, dialogue management, and natural language generation [1,7,8,20], which are designed separately. To reduce the human effort required to design and maintain these separate modules, recent work is changing from pipeline solutions [2, 4,21,22] to end-to-end dialogue models [3,23–25] that input plain texts and directly output a system response [5,26,27]. These end-to-end dialogue models focus on incorporating external knowledge bases into the learning framework. Eric et al. [9] proposed an end-to-end dialogue model that uses key–value pairs to represent the knowledge base while considering the one-hop relations of entities in their model. Madotto et al. [13] used disordered memory [28] to represent the knowledge triplets in the knowledge base. Wen et al. [10] combined the dialogue model with an attention mechanism, which retrieves entities from the knowledge base. Reddy et al. [29] utilized multi-level memory to represent the knowledge base in the form of queries, queried entities, and key–value pairs. GLMP (Global-to-Local Memory Pointer Networks) proposed by Wu et al. [12] estimates the tokens that will appear in the system response before the decoding process. Qin et al. [30] proposed a retriever to ensure that the predicted entities are in the same row of the tabular KB. Qin et al. [5] proposed a shared-private model to learn the shared features between different domains. Besides, many other efforts employed multi-task learning frameworks to jointly generate dialogue states and template responses. Lei et al. [31] proposed a two-stage copy mechanism for jointly generating dialogue states and system responses in an encoder–decoder architecture. Mehri et al. [32] proposed structured fusion networks to fuse the structural features of the pipeline solution into the end-to-end dialogue model. Zhang et al. [33] enhanced the decoder with multiple target responses to handle the one-to-many problem. However, the previous work ignored the graph structure information in the knowledge base and the relevant information between the knowledge base and the dialogue history. In contrast to the previous work, we build a heterogeneous relational graph to capture the graph structure information and jointly encode the two input texts.

2.2. Graph Neural Network

Graph Neural Network (GNN) is a kind of graph-based deep learning method [19,34–37]. This method has received much attention for its convincing performance and high scalability. Kipf et al. [34] proposed Graph Convolutional Network (GCN) for semi-supervised learning, which is a graph encoder based on convolutional neural networks and graph structures. Graph Attention Network (GAT) proposed by Veličković et al. [35] is a combination of GNN and the attention mechanism [38], which aims to focus on important neighbouring nodes. Zhang et al. [39] proposed cardinality preserved attention models to improve the performance of GAT. Schlichtkrull et al. [16] proposed Relational Graph Convolutional Network (RGCN) with block and basic decompositions to handle multi-relational graphs. Busbridge et al. [17] proposed Relational Graph Attention Network (RGAT) to explore the attention mechanism between neighbouring nodes under the same relation. Qi et al. [36] combined residual networks [40] with GCN to handle the face clustering task. Wang et al. [41] built a heterogeneous graph to combine word-level and sentence-level nodes and they adopted GAT in their task. Hong et al. [42] combined the attention mechanism with GNN to aggregate multi-relational information of heterogeneous vertices. Wang et al. [18] added a semantic-level attention mechanism to GAT to handle heterogeneous graphs. In contrast to the previous work, we propose Heterogeneous Relational Graph Neural Networks (HRGNN) that jointly encode the knowledge base and the dialogue history. In addition, our graph encoder contains two improvements for dealing with overfitting and confusion problems.

2.3. Class imbalance

To the best of our knowledge, we are the first to handle the entity imbalance problem in task-oriented dialogues, which can be regarded as a kind of class imbalance. To handle the class imbalance problem, the focal loss proposed by Lin et al. [43] reweights the losses of different categories to focus on hard and misclassified examples. Shan et al. [7] adjusted the learning weights according to the accuracy of each slot to handle the class imbalance problem in the slot-filling task. However, previous work mainly focused on simple classification tasks or slot-filling tasks rather than generation tasks. Lin et al. [43] adopted static learning weights in the focal loss for each category, which affects the performance on dialogue generation tasks that have numerous categories. The method proposed by Shan et al. [7] fails to distinguish between words and entities, limiting its performance on dialogue generation tasks. In our work, we propose an adaptive objective to evaluate the difficulty of different kinds of target entities, and then adaptively adjust the learning weights of different categories (i.e., different entities and words) to address the entity imbalance problem in dialogue generation tasks.

3. Task formulation

In this section, we will present the inputs and outputs of an end-to-end task-oriented dialogue system to facilitate the understanding of the task. At each dialogue turn, the dialogue system takes the dialogue history and the knowledge base as inputs and aims to output the system response. We denote the user utterance as u and the system response as s, thus the k-turn dialogue history can be represented as $\{D = (u^1, s^1), (u^2, s^2), \dots, u^k\}$. Moreover, the knowledge base can be represented as a collection of triplets. For tabular knowledge bases, we use table headers as relations to construct triplets, such as (“Stazione”, “food”, “Italian”). The triplets in the knowledge base form a directed relational graph, where entities correspond to nodes and relations correspond to edges. We represent the knowledge graph as $G = (\mathcal{V}; \mathcal{E}; \mathcal{R})$, where $v_i \in \mathcal{V}$ is an entity, $(v_i; r; v_j) \in \mathcal{E}$ is a triplet, and $r \in \mathcal{R}$ is a relation type.

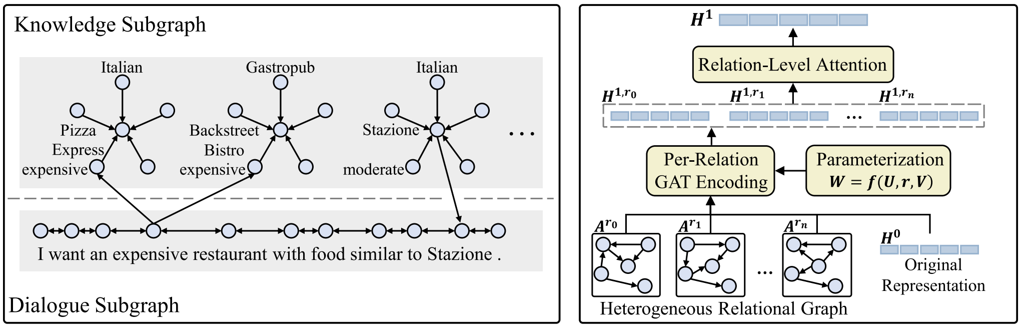
We define the end-to-end task-oriented dialogue task [9] as predicting the system response y according to the dialogue history D and the knowledge base G. Formally, the task is defined as:

$$\boldsymbol{p}(y|D, G) = \prod_{i=1}^{d} \boldsymbol{p}(y_i|y_1, \dots, y_{i-1}, D, G), \tag{1}$$

where y_i is the ith output token of the system response and d is the number of tokens.

4. Method

In order to handle the end-to-end task-oriented dialogue task, we propose the Heterogeneous Relational Graph Neural Networks with Adaptive Objective, which is illustrated in Fig. 2. The method consists of four important parts: Heterogeneous Relational Graph Construction, Heterogeneous Relational Graph Neural Networks, HRGNN-Based Task-Oriented Dialogue System, and Adaptive Objective. The heterogeneous relational graph jointly encodes the knowledge base and the dialogue history to capture the graph structure information. The heterogeneous relational graph neural networks adopt the shared-private parameterization and the hierarchical attention mechanism to encode the constructed graph until equilibrium. We integrate HRGNN into a task-oriented dialogue system. To train the dialogue system, an adaptive objective is proposed to handle the entity imbalance problem.



(A) Heterogeneous Relational Graph Construction

(B) Heterogeneous Relational Graph Neural Networks

Fig. 2. Overview of the heterogeneous relational graph (A) and the heterogeneous relational graph neural networks (B). For better understanding, we simplify some nodes and edges in the graph.

4.1. Heterogeneous relational graph construction

In this subsection, we describe how to construct the heterogeneous relational graph. In the heterogeneous relational graph, we use two kinds of basic units as nodes: words in the dialogue history and entities in the knowledge base. Entities are usually composed of multiple words. Therefore, in our method, the term "heterogeneous" implies that our method is able to handle different kinds of nodes, including word-level and entity-level nodes. Based on these nodes, we consider multi-perspective relations to construct effective paths for propagating relevant information. We will first introduce how to construct knowledge subgraphs on the basis of a tabular knowledge base. Next, we describe the construction of a dynamic dialogue subgraph. Finally, we describe the cross-subgraph relations that capture relevant information between the knowledge base and the dialogue history.

Knowledge Subgraph To capture structural information from the knowledge base, we should propagate the relevant information from the knowledge base (including relevant entities and corresponding relations) into each entity representation. As shown in Fig. 2, we propagate the attributes "moderate" and "Italian" into the entity "Stazione". In the tabular knowledge bases of the task-oriented dialogue datasets, we find that each row usually represents a single individual, such as a restaurant, and that there is no significant correlation between entities in different rows. Irrelevant information from other rows may negatively affect the entity representation and degrades performance. Therefore, for each entity, the knowledge subgraph should capture the structural information within each row to obtain an information-rich entity representation. With this guidance, we construct the knowledge subgraphs as follows: (1) We first represent each row of the knowledge base as multiple triplets. (2) We then construct a subgraph for each row to capture the graph structure information within the row. Specifically, if two entities (v_i, v_j) in the same row have a relation r, we use that relation as a directed edge to link the tail entity to the head entity. In addition, we use the inverse relation to connect the two entities from opposite directions. For example, we link the two entities "Stazione" and "Italian" through the two relations "food" and "is the food of", which are represented as two opposite edges in the graph. The adjacent matrix $\boldsymbol{A}_{ij}^r$ of the relation r in the knowledge base is defined as:

$$\boldsymbol{A}_{ij}^{r} = \begin{cases} 1 & \text{if } v_i \text{ and } v_j \text{ are in the same row and there} \\ & \text{is a relation } r \text{ between them,} \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Fig. 2(A) shows an example of the knowledge subgraphs. In these subgraphs, the entities within each row are connected according to their relations in the knowledge base. The different rows are connected by the dialogue history. Therefore, each entity can obtain structural information from its corresponding row and other rows related to the dialogue topic. We also tried a very plain way to construct the knowledge subgraph, which transforms the whole knowledge base into a large knowledge graph. This way degrades performance because it allows the propagation of irrelevant information within the large knowledge graph.

Dialogue Subgraph In order to receive the knowledge information in the knowledge subgraph and to propagate the knowledge information into the word representation of the dialogue history, it is necessary to construct a dialogue subgraph. Due to the complexity of dialogues, it is infeasible to accurately identify the knowledge information required for each word and construct a specific dialogue subgraph. Therefore, in our work, we employ an effective dynamic construction strategy, which links every word to each other through two relations, "forward" and "backward". For example, the adjacent matrix of the relation "forward" is defined as:

$$\boldsymbol{A}_{ij}^{forward} = \begin{cases} 1 & \text{if the word } w_j \text{ is in front of the word } w_i, \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

where w_i and w_j are two words in the dialogue history. After obtaining the relevant knowledge from the knowledge subgraphs, the dynamic dialogue subgraph spreads it easily to each word representation due to the full connection. In addition, our method allocates different weights to different nodes and relations through attention mechanisms to learn the importance of different knowledge information. In our work, we tried some partial connections based on prior knowledge to construct dialogue subgraphs, e.g., the co-reference relation or the keyword relation. These subgraphs usually only focus on only a few specific aspects and tend to introduce unexpected biases.

Cross-Subgraph Relations Last but not least, we utilize cross-subgraph relations to capture relevant information between the knowledge base and the dialogue history. In our model, we link the co-occurring entities in the two kinds of subgraphs through two relations, "Knowledge to Dialogue" and "Dialogue to Knowledge". For example, the adjacent matrix of the relation "Knowledge to Dialogue" is defined as:

$$\boldsymbol{A}_{ij}^{K2D} = \begin{cases} 1 & \text{if } w_i \text{ in the dialogue history is an entity } v_j \\ & \text{in the knowledge base,} \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

In the data preprocessing phase, we connect conversational words belonging to the same entity as a whole token to match entities in the knowledge base. The two relations provide paths for capturing cross-subgraph relevant information. As shown in Fig. 2(A), we link the restaurant "Stazione" in the knowledge subgraph to the restaurant "Stazione" in the dialogue subgraph through the two relations. As a result, there is a relational path that propagates the attribute "Italian" of the restaurant "Stazione" into the dialogue. The relational path also propagates the dialogue information into the knowledge base. In addition, different knowledge subgraphs are connected by the entities mentioned in the dialogue to obtain accurate entity representations.

To propagate information from a node to itself, we link each node to itself via a relation "self-loop". The adjacent matrix of the relation "self-loop" is a diagonal matrix.

In conclusion, we constructed a heterogeneous relational graph by capturing multi-perspective relations between the knowledge base and the dialogue history. This graph jointly encodes the two input texts to capture the natural graph structure information in the knowledge base. Based on the dialogue information, it allows the entities to focus on dialogue-related attributes. In addition, it facilitates dialogue understanding by accessing the knowledge base. In this way, we can effectively incorporate the knowledge base into the learning framework.

4.2. Heterogeneous relational graph neural networks

Since the heterogeneous relational graph contains a large number of imbalanced relations, it is suboptimal to directly deploy existing RGNNs [16–18] on the constructed graph. The previous RGNNs suffer from overfitting and confusion problems. We propose a shared-private parameterization and a hierarchical attention mechanism to alleviate these two problems. More specifically, we define the following propagation model to calculate the updates of the nodes in the graph:

$$\boldsymbol{h}_i^{l+1} = \sigma\Big(\sum_{r\in\mathcal{R}} c^{r,l}\sigma\Big(\sum_{j\in\mathcal{N}_i^r} a_{ij}^{r,l}\boldsymbol{W}^{r,l}\boldsymbol{h}_j^l\Big)\Big), \tag{5}$$

where σ is the activation function. $c^{r,l}$ is the attention weight for the relation r. $\mathcal{N}_i^r$ denotes the set of neighbouring nodes of the node i under the relation r. $a_{ij}^{r,l}$ is the attention weight between two nodes i and j under the relation r. $\boldsymbol{W}^{r,l}$ is a relation-specific trainable matrix that projects hidden states into the relation feature space. Intuitively, the graph encoder accumulates the transformed features of neighbouring nodes by normalized summations. In practice, with the adjacent matrix ($\boldsymbol{A}^r$), the graph encoder can be efficiently implemented using sparse matrix multiplications to avoid explicit summations and allow parallel computation of each node [16]. The original representation of the conversational words is obtained from the underlying encoding module of the task-oriented dialogue system. And, we represent each entity as a vector via the underlying embedding function. We denote the original representation of all nodes as $\boldsymbol{H}^0 = (\boldsymbol{h}_1^0, \boldsymbol{h}_2^0, \ldots, \boldsymbol{h}_{n'}^0) \in \mathbb{R}^{n'\times d^0}$, where n' is the number of nodes and d^0 is the dimension of the original representation.

In Eq. (5), $\boldsymbol{h}_j^l$ contains hidden representations of heterogeneous nodes, including words in the dialogue history and entities in the knowledge base. These different types of nodes are represented as individual nodes. In addition, we adopt relation-specific matrices $\boldsymbol{W}^{r,l}$ in Eq. (5) to handle multi-relational information. The relation-specific matrices are constructed by the proposed shared-private parameterization. Therefore, we call our method heterogeneous relational graph neural networks.

4.2.1. Shared-private parameterization

In the graph encoder, the relation-specific matrixes ($\boldsymbol{W}^{r,l}$) increase linearly with the number of relations. Since the heterogeneous relational graph contains a large number of imbalanced relations, it is easy to overfit the model to rare relations. To alleviate this problem, there are two previous approaches: block decomposition and basis decomposition [16]. However, they are either incapable of learning shared features from high-frequency relations or incapable of maintaining the private features of each relation. Therefore, we propose the shared-private parameterization to alleviate the overfitting problem by balancing shared features and private features, which integrates the advantages of existing works. In this subsection, we will first analyse the shortcomings of existing works and then introduce our method.

The block decomposition represents $\boldsymbol{W}^{r,l}$ as a combination of block-diagonal matrixes:

$$\boldsymbol{W}^{r,l} = \mathrm{diag}(\boldsymbol{Q}_1^{r,l}, \ldots, \boldsymbol{Q}_b^{r,l}), \tag{6}$$

where $\boldsymbol{Q}_1^{r,l} \in \mathbb{R}^{(d^{l+1}/b)\times(d^l/b)}$ is a low-dimensional matrix. d^l is the dimension of the node representation in the lth layer. The block decomposition adds a sparsity constraint on $\boldsymbol{W}^{r,l}$. It does not contain shared parameters. The independent relation matrixes assume that the relation spaces are independent, making this approach incapable of learning efficient features from high-frequency relations.

The basis decomposition represents $\boldsymbol{W}^{r,l}$ as a linear combination of shared parameters:

$$\boldsymbol{W}^{r,l} = \sum_{i=1}^{b} e_i^{r,l}\boldsymbol{Z}_i^l, \tag{7}$$

where $\boldsymbol{Z}_i^l \in \mathbb{R}^{d^{l+1}\times d^l}$ is a shared trainable matrix and $e_i^{r,l}$ is a private trainable parameter of the relation r. Since the private parameters ($e^{r,l}$) are so few compared to the shared parameters, this approach cannot effectively capture the private features of each relation.

In this paper, we propose a shared-private parameterization to alleviate the overfitting problem, which decomposes $\boldsymbol{W}^{r,l}$ as:

$$\boldsymbol{W}^{r,l} = \boldsymbol{U}^l ø^{\mathrm{emb}}(r)(\boldsymbol{V}^l)^{\mathrm{T}}, \tag{8}$$

where $ø^{\mathrm{emb}}(*)$ is an embedding function. $\boldsymbol{U}^l \in \mathbb{R}^{d^{l+1}\times d^r}$ and $\boldsymbol{V}^l \in \mathbb{R}^{d^l\times d^r}$ are two trainable matrixes shared by all relations. The proposed shared-private parameterization is inspired by the Singular Value Decomposition (SVD). The private relation embedding vector is similar to the singular value. The two shared matrixes seem like the unitary matrix. Here, we use the private relation embedding vector and the two shared matrixes to construct a relation-specific matrix, but do not actually perform the SVD. The shared matrixes ($\boldsymbol{U}^l$, $\boldsymbol{V}^l$) learn the shared features between relations to alleviate the overfitting problem. The private parameters preserve the private features of each relation. Compared to the previous work [16–18], our method is a better way to integrate shared and private parameters, which eventually exhibits higher performance. The shared-private mechanism is usually adopted in multi-task learning [44,45]. In our method, we utilize this mechanism to construct the relation-specific matrixes.

4.2.2. Hierarchical attention mechanism

As mentioned earlier, previous RGNNs usually treat all relations as equally important, which makes it difficult for them to capture important relational information from a large number of relations. The attention mechanism [38] is widely used in many other natural language processing tasks, which selects important information by assigning different weights to different units. In our work, we propose a hierarchical attention mechanism, which contains a node-level attention mechanism and a relation-level attention mechanism to capture important information.

As shown in Eq. (5), the node-level attention mechanism assigns different weights to different nodes under the same relation. By amplifying or minimizing the hidden state, it accumulates important information of neighbouring nodes under a specific relation into the current node representation. The node-level attention mechanism calculates the weights as follows:

$$a_{ij}^{r,l} = \frac{\exp((\boldsymbol{W}^{r,l}\boldsymbol{h}_i^l)^{\mathrm{T}}(\boldsymbol{W}^{r,l}\boldsymbol{h}_j^l))}{\sum_{k\in\mathcal{N}_i^r}\exp((\boldsymbol{W}^{r,l}\boldsymbol{h}_i^l)^T(\boldsymbol{W}^{r,l}\boldsymbol{h}_k^l))}. \tag{9}$$

The relation-level attention mechanism uses a trainable parameter ($c^{r,l}$) to assign different weights to different relations:

$$c^{r,l} = \frac{\exp(o^{r,l})}{\sum_{r'\in\mathcal{R}}\exp(o^{r',l})}, \tag{10}$$

where $o^{r,l}$ is a non-normalized parameter for the relation r. The relation-level attention mechanism allows the heterogeneous relational graph neural networks to pay more attention to the node representations under important relations. The hierarchical attention mechanism is inspired by other work [18,46]. Here, we design the hierarchical attention mechanism to encode the heterogeneous relational graph.

In general, we stack multiple layers to capture dependencies across multiple relational steps. In this way, our model captures long-term dependencies in the heterogeneous relational graph.

4.3. HRGNN-based task-oriented dialogue system

In this section, we introduce how to integrate the heterogeneous relational graph neural networks into a task-oriented dialogue system. The dialogue system consists of Graph-Based Encoding and Graph-Based Decoding, which is shown in Fig. 3.

4.3.1. Graph-based encoding

In the graph-based encoder, we first connect the entire dialogue history $(u^1, s^1, u^2, s^2, \dots, u^k)$ word by word and represent it as $X = (x_1, x_2, \dots, x_n)$, where n is the number of conversational words. We use a Bidirectional Gated Recurrent Unit (BiGRU [47]) to encode X as: $\boldsymbol{h}_i^{0,x} = \mathrm{BiGRU}(\o^{\mathrm{emb}}(x_i), \boldsymbol{h}_{i-1}^{0,x})$ to obtain the original context-sensitive representation of the words. To obtain the original representation of the entities, we first convert each entity into a vector using the entity embedding function. Then, we sum each entity vector with the vectors of manually specified relevant knowledge as its original representation, which proves to be an effective way to enhance the entity representation [5,12].

The heterogeneous relational graph neural networks take the original representation and the constructed graph as inputs to jointly encode the knowledge base and the dialogue history. After multi-layer propagation, we can obtain the graph-based node representation, which contains a large amount of node-level and relation-level information. To incorporate the graph-based node representation into the decoding process, we treat the graph-based representation as supporting information and sum them with the original representation in the memory networks.

4.3.2. Graph-based decoding

Different from the typical sequence-to-sequence models [48,49], a successful task-oriented dialogue system relies heavily on an accurate knowledge retriever. We utilize the global-to-local pointer mechanism [12] as our knowledge retriever, which shows the best performance.

To obtain the initial state of the response decoder (i.e., the GRU decoder), we perform multi-hop reasoning in the memory networks to summarize the graph information. At the kth hop, the reasoning process is computed as follows:

$$\boldsymbol{p}_i^k = \frac{\exp((\boldsymbol{q}^k)^{\mathrm{T}}\boldsymbol{m}_i^k)}{\sum_{j\in\mathcal{N}}\exp((\boldsymbol{q}^k)^{\mathrm{T}}\boldsymbol{m}_j^k)}, \tag{11}$$

$$\boldsymbol{o}^k = \sum_{i=0}^{\mathcal{N}}\boldsymbol{p}_i^k\boldsymbol{m}_i^{k+1}, \tag{12}$$

$$\boldsymbol{q}^{k+1} = \boldsymbol{o}^k + \boldsymbol{q}^k, \tag{13}$$

where $\boldsymbol{m}_i^k$ is the hidden state of the ith node in the kth memory layer. $\mathcal{N}$ is the number of nodes. The initial query vector $\boldsymbol{q}^1$ is $\boldsymbol{h}_n^{0,x}$. After k-hop reasoning, $\boldsymbol{q}^{k+1}$ can be treated as summarized graph information and used to initialize the decoder. Following the previous work [12], we compute a global pointer $\boldsymbol{p}_{\mathrm{global}}$ as: $\boldsymbol{p}_{\mathrm{global},i} = \mathrm{sigmoid}((\boldsymbol{q}^k)^{\mathrm{T}}\boldsymbol{m}_i^k)$, where sigmoid($*$) is the activation function. The global pointer is used to estimate the nodes that will appear in the system response.

The GRU decoder recurrently predicts the output token y_i by decoding the hidden state $\boldsymbol{h}_i^{\mathrm{dec}}$. The hidden state is projected into the vocabulary space as follows:

$$\boldsymbol{p}(y_i|y_1, \dots, y_{i-1}, D, G) = \mathrm{softmax}(\boldsymbol{W}^v[\boldsymbol{h}_i^{\mathrm{dec}}, \boldsymbol{h}_i^a]), \tag{14}$$

where softmax($*$) is the normalized function and $\boldsymbol{W}^v$ is the trainable matrix. $\boldsymbol{h}_i^a$ is the hidden state obtained by the attention mechanism [50] on the nodes in the dialogue subgraph. The vocabulary probability is used to predict words as output tokens. In addition, we use a local pointer to retrieve nodes from the graph as output tokens. The local pointer uses the hidden state $[\boldsymbol{h}_i^{\mathrm{dec}}, \boldsymbol{h}_i^a]$ as the initial query vector $\boldsymbol{q}_{\mathrm{local}}^1$ to perform multi-hop reasoning on the graph-based memory layers. The probability of the local pointer is calculated as follows:

$$\boldsymbol{p}_{\mathrm{local},i} = \frac{\exp((\boldsymbol{q}_{\mathrm{local}}^k)^{\mathrm{T}}\boldsymbol{m}_i^k\boldsymbol{p}_{\mathrm{global},i})}{\sum_{j\in\mathcal{N}}\exp((\boldsymbol{q}_{\mathrm{local}}^k)^{\mathrm{T}}\boldsymbol{m}_j^k\boldsymbol{p}_{\mathrm{global},j})}. \tag{15}$$

When a placeholder (e.g., "@food") is generated from the vocabulary, we choose the node with the highest probability in $\boldsymbol{p}_{\mathrm{local}}$ as the output token.

4.4. Adaptive objective

In general, we can train the model with the cross-entropy loss between the three probabilities ($\boldsymbol{p}(y|D, G)$, $\boldsymbol{p}_{\mathrm{global}}$, $\boldsymbol{p}_{\mathrm{local}}$) and their targets. In this paper, we take into account the entity imbalance problem in task-oriented dialogues. The entity imbalance problem can be viewed as a class imbalance problem since there is an imbalance between different kinds of entities. Instead of treating all entities indiscriminately, we propose a novel adaptive objective to balance the learning of different kinds of target entities.

The adaptive objective evaluates the difficulty based on the micro F1 of each kind of entities on the validation set and adaptively adjusts the weight of each kind of entities during the optimization process. We denote the micro F1 of a kind of entities on the validation set as F. If $F_t < F_k$, our model achieves poorer performance on the class t. We can argue that the entities belonging to the class t are more difficult than the entities belonging to the class k. Therefore, we increase the loss weight of the class t and decrease the loss weight of the class k to allow the model to focus more on learning the difficult class t. Specifically, if the ith token in the target response is an entity, its loss weight α_i is defined as:

$$\alpha_i = \frac{1 - F_i}{\sum_{j\in\mathcal{R}} 1 - F_j}|\mathcal{R}| + \beta \tag{16}$$

where β is a hyper-parameter. $|\mathcal{R}|$ is the number of entity classes. Otherwise, if the ith token in the target response is a word, its weight α_i is β. In fact, β is the basic weight used to train the dialogue system. As shown in Eq. (16), we assign higher loss weights to entities in difficult categories, which allows the model to focus on learning entities in that category. The adaptive objective is defined as follows:

$$\mathcal{L} = -\sum_{i=1}^{d}\alpha_i\log(\boldsymbol{p}(y_i|y_{<i}, D, G)) \tag{17}$$

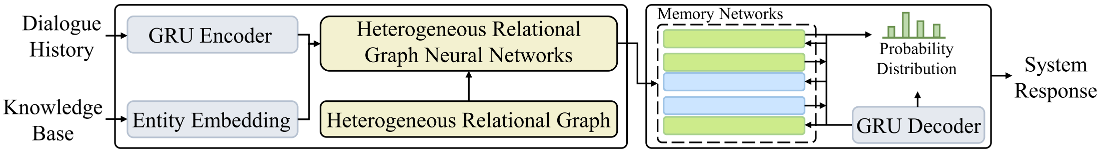


Fig. 3. Architecture of a task-oriented dialogue system based on heterogeneous relational graph neural networks.

Table 1
Statistics of datasets. We list the number of examples, the number of relations, and the number of nodes.

Dataset	Train	Valid	Test	Relations	Nodes
SMD	6290	777	807	67	1490
Extended Multi-WOZ 2.1	8529	576	711	25	3605

The adaptive objective fits the data better by dynamically evaluating the difficulty in a performance-sensitive manner. When an epoch ends, the adaptive objective re-evaluates the difficulty of each kind of entity and updates F to ensure that the loss of hard entities is larger than that of simple entities. Thus, it dynamically balances the learning of all entities during the training process.

5. Experiments

In this section, we construct a series of experiments to demonstrate the effectiveness of our proposed method. The first of them is to evaluate the overall performance of the HRGNN-Based task-oriented dialogue system. Next, we verify whether each of our proposed components, including the heterogeneous relational graph, the shared-private parameterization, the hierarchical attention mechanism, and the adaptive objective, can effectively solve its corresponding problems. Finally, we manually evaluate these task-oriented dialogue systems to give fair evaluations. We list the research questions (RQ) that guide our experiments.

- RQ1. What is the overall performance of our model in end-to-end task-oriented dialogues?
- RQ2. Is the heterogeneous relational graph effective for end-to-end task-oriented dialogues?
- RQ3. Is the shared-private parameterization effective for handling the overfitting problem?
- RQ4. Is the hierarchical attention mechanism effective for handling the confusion problem?
- RQ5. Can the adaptive objective improve performance through balanced training?
- RQ6. Can our model achieve greater improvements in dialogues that rely on rich knowledge?
- RQ7. Can our model outperform previous models in human evaluation?
- RQ8. Is our model still effective when using other underlying encoders?

In the remainder of this section, we first introduce the datasets and the baselines. Then, we show the experimental settings. Finally, we show the results and analysis.

5.1. Datasets

We evaluate the proposed approach on the SMD dataset [9] and the extended Multi-WOZ 2.1 dataset [5,14,15]. Based on the Multi-WOZ 2.1 dataset [15], Qin et al. [5] proposed the extended Multi-WOZ 2.1 dataset, where each dialogue is equipped with a corresponding knowledge base. The two datasets contain dialogues from multiple domains. The SMD dataset is designed for car assistants, which contains three domains: navigation, weather, and calendar. The extended Multi-WOZ 2.1 dataset contains three domains: restaurant, attraction, and hotel. Detailed statistics are shown in Table 1. We follow the same partitions in the datasets as the previous work [5,12,14]. Due to the knowledge-rich nature and the flexible expressions, it is extremely challenging to develop dialogue systems on the two datasets. To save space, we follow the previous work [5] and call the extended Multi-WOZ 2.1 dataset as the Multi-WOZ 2.1 dataset. The two task-oriented dialogue datasets provide a specific knowledge base for each dialogue, so we do not need to use external knowledge base datasets.

Table 2
Hyper-parameters for the SMD and Multi-WOZ 2.1 datasets.

Hyper-parameters	SMD	Multi-WOZ 2.1
Graph layers	2	2
Graph hidden size	128	128
Batch size	16	16
Hidden size	128	128
Embedding size	128	128
Learning rate	0.001	0.001
Dropout rate	0.2	0.0
Teacher forcing rate	0.9	0.9
Memory network layers	3	3
β	1.0	1.0

5.2. Baselines

We compare our model with the following state-of-the-art baselines.

- **Mem2Seq** [13]: The model separately encodes conversational words and knowledge triplets in memory networks. It uses a pointer network to retrieve entities for the response.
- **DSR** [10]: The model uses implicit dialogue states to enhance the ability of knowledge retrieval.
- **KB-Retriever** [30]: The model utilizes a special retriever to ensure that the predicted entities are in the same row of the knowledge base.
- **GLMP** [12]: The model retrieves entities through the global-to-local pointer mechanism, which can filter out the information that appears in the system response.
- **DFNet** [5]: The model uses shared-private parameters to explore the relations between different domains.

5.3. Experimental settings

Following the previous work [5], we use the Adam optimizer [51] to train our model with an initial learning rate of 0.001. The learning rate decays by half every 2 epochs if there is no improvement on the validation set. The dimensional size of the embedding layer and the graph layers is 128. To improve generalization, we adopt a dropout ratio from [0.0, 0.1, 0.2, 0.3],

Table 3
Main results. The numbers with † indicate that the improvement of our method is statistically significant with $p < 0.01$ under the t-test over the previous strong models, GLMP and DFNet. The numbers with * indicate that the improvement of our method is statistically significant with $p < 0.05$.

Model	SMD					Multi-WOZ 2.1				
	BLEU	F1	Navigate F1	Weather F1	Calendar F1	BLEU	F1	Restaurant F1	Attraction F1	Hotel F1
Mem2Seq [13]	12.6	33.4	20.0	32.8	49.3	6.6	21.62	22.4	22.0	21.0
DSR [10]	12.7	51.9	52.0	50.4	52.1	9.1	30.0	33.4	28.0	27.1
KB-retriever [30]	13.9	53.7	54.5	52.2	55.6	–	–	–	–	–
GLMP [12]	13.9	60.7	54.6	56.5	72.5	6.9	32.4	38.4	24.4	28.1
DFNet [5]	14.4	62.7	57.9	57.6	73.1	9.4	35.1	40.9	28.1	30.6
HRGNN-AO	**16.5**†	**65.2**†	**58.6**†	**62.8***	**75.1**†	**10.1***	**38.0**†	**41.7**†	**33.8***	**35.2**†

Table 4
Effectiveness of the knowledge subgraph.

Model	SMD (BLEU)		SMD (Entity F1)		Multi-WOZ 2.1 (BLEU)		Multi-WOZ 2.1 (Entity F1)	
	Test	Δ	Test	Δ	Test	Δ	Test	Δ
HRGNN-AO	**16.5**	–	**65.2**	–	**10.1**	–	**38.0**	–
with A Large KG	15.7	−0.8	64.1	−1.1	9.8	−0.3	37.2	−0.8
w/o Knowledge Subgraph	15.3	−1.2	63.4	−1.8	9.5	−0.6	36.1	−1.9

which is the same as the previous work [5]. We did not observe further improvements when using other dropout ratios. The batch size is selected from [16, 32] because we could deploy the model under these batch sizes on a single GPU (GeForce RTX 2080Ti) and obtain acceptable runtimes. We found that the entities in the system responses are often within two hops from the entities mentioned in the dialogue history. Therefore, we explore the number of graph layers from [1, 2, 3, 4]. All hyper-parameters are selected based on the validation set through a grid search. We show more details in Table 2.

For the task-oriented dialogue systems, the Micro Entity F1 metric is used to evaluate whether the predicted system responses contain accurate entities. Besides, we use BLEU [52] to evaluate the language quality of the predicted system responses.

5.4. Main results

We address the RQ1 in this subsection. We conduct experiments on the two datasets to compare our proposed method with the above baselines. The results are shown in Table 3. Since Wu et al. [12] reported Macro Entity F1 as Micro Entity F1, Qin et al. [5] reran the GLMP model and reported the new results in their paper. Therefore, we follow Qin et al. [5] and show the new results in Table 3. We also list separate results for different domains in this table. From the results, we can observe that:

(1) Our method outperforms the baselines by a large margin on both datasets. On the SMD dataset, compared with the model DFNet, our method achieves 2.1% and 2.5% improvements in terms of BLEU and Entity F1, respectively. It indicates that our method can generate more fluent responses and our method has a better ability to retrieve accurate entities. On the Multi-WOZ 2.1 dataset, we observe the same trend of improvement, i.e., our model improves by 0.7% and 2.9% on BLEU and entity F1 over the previous best model. Our method is more effective for task-oriented dialogues than the baselines.

(2) For each domain in the SMD dataset, our method achieves significant improvements of 0.7%, 5.2%, and 2.0% in terms of entity F1. The experimental results show that our method is effective and scalable in multiple domains. On the Multi-WOZ 2.1 dataset, our method also achieves significant improvements of 0.8%, 5.7%, and 4.6% in terms of entity F1. It indicates that these domains all suffer from two problems: how to effectively incorporate the knowledge base into the learning framework and the entity imbalance problem. Therefore, our method improves performance in these domains, which demonstrates the scalability of our method.

5.5. Effectiveness of our method

We study the advantages of our method from several aspects. First, we perform several experiments to analyse the effect of the heterogeneous relational graph. Next, we verify the effectiveness of the heterogeneous relational graph neural networks. Finally, we evaluate the validity of the adaptive objective to understand how it improves performance.

5.5.1. Strategy of building the heterogeneous relational graph

In this subsection, we assess the RQ2. To gain more insights into the structure of the heterogeneous relational graph, we explore different strategies to build the graph.

Table 4 shows the results of the models using different strategies to construct the knowledge subgraph. When representing the knowledge base as a large knowledge graph (“with A Large KG”), the model achieves a performance drop by 1.1% and 0.8% in terms of entity F1. In addition, this model reduces the language quality with a drop by 0.8% and 0.3% on BLEU. The results are consistent with our description in Section 4.1. For tabular KBs in task-oriented dialogue datasets, irrelevant information from other rows adversely affects the entity representation and ultimately degrades performance. In our model, only knowledge rows with entities mentioned in the dialogue history are connected. These rows are usually related to the dialogue topic and contain relevant information to other rows. When removing the knowledge subgraph (“w/o Knowledge Subgraph”), the model achieves a significant drop by 1.8% and 1.9% in terms of entity F1. It shows that the knowledge subgraph is critical for capturing the graph structure information.

Table 5 shows the results of the models with different strategies to construct the dialogue subgraph. When the dialogue history is represented as partial connection graphs, the performance of the model is significantly degraded. For “with Keyword DG”, we construct a dialogue subgraph that links entities, relations, and pronouns mentioned in the dialogue history to capture keyword information. For “with Entity DG”, we construct dialogue subgraphs that use relations from the knowledge base to link entities mentioned in the dialogue history. The two variants achieve low performance. It shows that these partial connections, constructed based on prior knowledge, are prone to introduce unexpected biases and are not sufficiently representative. When removing the dialogue subgraph (“w/o DG”), the model achieves a significant drop by 1.8% and 1.6% in terms of entity F1. The dynamic dialogue subgraph is effective in receiving knowledge information and dynamically propagating the information to each

Table 5
Effectiveness of the dialogue subgraph and the heterogeneous relational graph.

Model	SMD (BLEU)		SMD (Entity F1)		Multi-WOZ 2.1 (BLEU)		Multi-WOZ 2.1 (Entity F1)	
	Test	Δ	Test	Δ	Test	Δ	Test	Δ
HRGNN-AO	**16.5**	–	**65.2**	–	**10.1**	–	**38.0**	–
with Keyword DG	16.1	−0.4	64.3	−0.9	9.7	−0.4	37.4	−0.6
with Entity DG	15.7	−0.8	63.7	−1.5	9.6	−0.5	36.8	−1.2
w/o DG	15.4	−1.1	63.4	−1.8	9.3	−0.8	36.4	−1.6
w/o Constructed Graph	14.6	−1.9	63.0	−2.2	8.7	−1.4	35.9	−2.1

Table 6
Effectiveness of the multi-perspective relations.

Model	SMD (BLEU)		SMD (Entity F1)		Multi-WOZ 2.1 (BLEU)		Multi-WOZ 2.1 (Entity F1)	
	Test	Δ	Test	Δ	Test	Δ	Test	Δ
HRGNN-AO	**16.5**	–	**65.2**	–	**10.1**	–	**38.0**	–
w/o Inverse Relation	15.6	−0.9	63.6	−1.6	9.5	−0.6	36.5	−1.5
w/o Knowledge to Dialogue	15.8	−0.7	64.0	−1.2	9.9	−0.2	37.6	−0.4
w/o Dialogue to Knowledge	16.2	−0.3	63.5	−1.7	9.3	−0.8	36.9	−1.1
w/o CG Relation	15.5	−1.0	63.3	−1.9	9.6	−0.5	36.4	−1.6

Table 7
Effectiveness of the shared-private parameterization.

Model	SMD (BLEU)		SMD (Entity F1)		Multi-WOZ 2.1 (BLEU)		Multi-WOZ 2.1 (Entity F1)	
	Test	Δ	Test	Δ	Test	Δ	Test	Δ
HRGNN-AO	**16.5**	–	**65.2**	–	**10.1**	–	**38.0**	–
with Block Decomposition	15.4	−1.1	64.0	−1.2	9.6	−0.5	37.2	−0.8
with Basic Decomposition	15.9	−0.6	63.8	−1.4	10.0	−0.1	37.4	−0.6
w/o Shared Parameters	15.2	−1.3	63.6	−1.6	9.3	−0.8	37.5	−0.5
w/o Private Parameters	14.7	−1.8	63.5	−1.7	9.5	−0.6	36.9	−1.1

word through the attention mechanism. Removing the heterogeneous relational graph ("w/o Constructed Graph") reduces the performance by 2.2% and 2.1% in terms of entity F1. It validates the effectiveness of the proposed graph.

We also experimentally study the effect of relations in the heterogeneous relational graph. The results are shown in Table 6. For "w/o Inverse Relation", we remove the inverse relation from the knowledge subgraph. The performance degradation verifies the effectiveness of propagating the graph structure information from two directions. Removing each of the two relations, "Knowledge to Dialogue" or "Dialogue to Knowledge", hurts performance because they provide important relational paths for propagating relevant information between the knowledge base and the dialogue history. For "w/o CG Relation", we remove the two relations, "Knowledge to Dialogue" and "Dialogue to Knowledge". The performance degradation verifies that it is effective to simultaneously exploit the two relations.

5.5.2. Effectiveness of shared-private parameterization

Next, we turn to the RQ3 in this subsection. To gain more insights into the proposed shared-private parameterization, we test many variants of the heterogeneous relational graph neural networks. The results are shown in Table 7. When we replace the proposed shared-private parameterization with the block or basic decomposition [16], the performance degrades significantly. The results show that the proposed shared-private parameterization is an effective way to deal with the overfitting problem by integrating shared and private parameters. Besides, for "w/o Shared Parameters", we test a variant that only adopts private matrixes. The performance of this variant degrades significantly. The results demonstrate the effectiveness of the shared parameters, which transfer shared features from high-frequency relations to low-frequency relations. For "w/o Private Parameters", the model only adopts the shared matrixes. This model achieves low performance because it treats all relations as the same, which hinders the learning of the private features of each relation.

5.5.3. Effectiveness of hierarchical attention mechanism

In this subsection, we assess the RQ4. Table 8 shows the results. To explore the importance of the proposed hierarchical attention mechanism, we remove each level of the hierarchical attention mechanism. On the SMD dataset, removing the relation-level attention mechanism ("w/o Relation-Level Attention") leads to a performance drop by 1.1% on BLEU and 0.9% on entity F1. The relation-level attention mechanism brings significant improvements by enabling HRGNN-AO to focus on important relational information. When removing the node-level attention mechanism ("w/o Node-Level Attention"), the performance drops significantly because this model does not pay attention to important nodes under the same relation. Furthermore, removing the hierarchical attention mechanism results in low performance. The results demonstrate that the hierarchical attention mechanism is effective in accumulating the node-level and relation-level structural information. Besides, we replace our graph encoder with the previous heterogeneous graph attention network [18] to our tasks, which contains node-level and semantic-level attention mechanisms. On the SMD dataset, it achieves 63.5% on Entity F1 and 15.6% on BLEU, which achieves lower performance. Due to the lack of shared-private parameterization, the heterogeneous graph attention network [18] suffers from weak generalization.

5.5.4. Effectiveness of adaptive objective

In this section, we assess the RQ5. The results are shown in Table 9. To explore the importance of adjusting the weight α_i adaptively, we remove the adaptive objective ("w/o Adaptive Objective"), which leads to a drop by 0.9% and 0.5% in terms of entity F1. Replacing the adaptive objective with the focal loss ($\gamma \in [2, 3]$, [43]) leads to a drop by 0.7% and 0.6% on entity F1. Compared with the original cross-entropy loss, the focal loss does not bring in significant improvements.

Fig. 4 shows the changes in entity F1 of each kind of entity on the SMD dataset between the adaptive objective and the original cross-entropy objective. We rank all entity classes in ascending order according to their frequency. Thus, entities

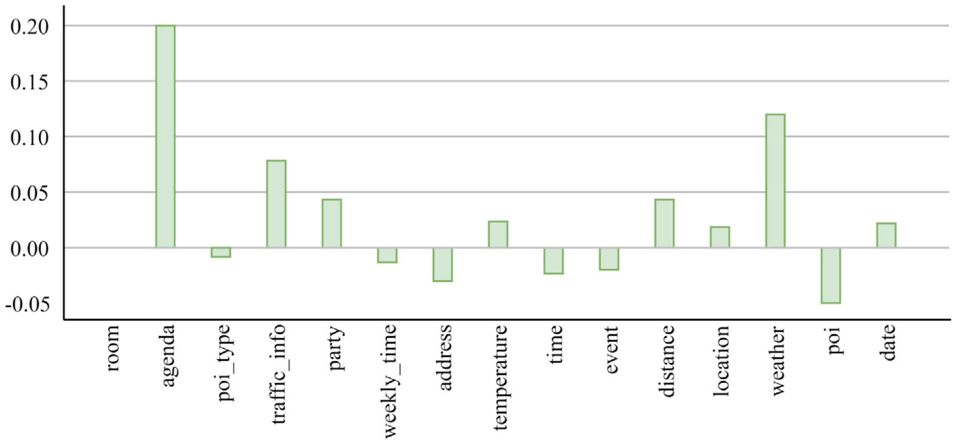


Fig. 4. Entity F1 changes between the adaptive objective and the original cross-entropy objective for each kind of entity on the SMD dataset. We rank all entity relations in ascending order according to their frequency.

Table 8
Effectiveness of the hierarchical attention mechanism.

Model	SMD (BLEU)		SMD (Entity F1)		Multi-WOZ 2.1 (BLEU)		Multi-WOZ 2.1 (Entity F1)	
	Test	Δ	Test	Δ	Test	Δ	Test	Δ
HRGNN-AO	**16.5**	–	**65.2**	–	**10.1**	–	**38.0**	–
w/o Relation-level attention	15.4	−1.1	64.3	−0.9	9.7	−0.4	37.4	−0.6
w/o Node-level attention	15.5	−1.0	63.7	−1.5	9.1	−1.0	36.8	−1.2
w/o HAM	15.2	−1.3	63.3	−1.9	8.9	−1.2	36.3	−1.7
with Heterogeneous GAT	15.6	−0.9	63.5	−1.7	9.6	−0.5	37.1	−0.9

Table 9
Effectiveness of the adaptive objective.

Model	SMD (BLEU)		SMD (Entity F1)		Multi-WOZ 2.1 (BLEU)		Multi-WOZ 2.1 (Entity F1)	
	Test	Δ	Test	Δ	Test	Δ	Test	Δ
HRGNN-AO	**16.5**	–	**65.2**	–	**10.1**	–	**38.0**	–
w/o Adaptive Objective	16.4	−0.1	64.3	−0.9	9.4	−0.7	37.5	−0.5
with Focal Loss [43]	16.3	−0.2	64.5	−0.7	9.8	−0.3	37.4	−0.6

Table 10
Human evaluation. “Average Agreement” is the inter-annotator agreement (Kappa Coefficient [53]).

Model	Correct	Fluent	Human-like
GLMP [12]	3.51	3.96	4.04
DFNet [5]	3.60	4.12	4.06
HRGNN-AO	**4.17**	**4.35**	**4.39**
Average Agreement	67.1%	59.8%	62.7%

belonging to the left part are relatively more difficult than those belonging to the right part. In the model with the adaptive objective, entities belonging to the left part achieve significant improvements, demonstrating that the adaptive objective can encourage the learning of low-frequency entity classes. While the adaptive objective tends to reduce the weight of entities belonging to the right part, these entities also benefit from the adaptive objective. We argue that this is because the adaptive objective learns all entities in a balanced way by dynamically evaluating and optimizing hard target entities, thereby improving the performance of many kinds of entities.

5.6. Dialogues relying on rich knowledge

In this subsection, we answer the RQ6. We believe that the entities mentioned in the dialogue history often require knowledge for better understanding. Therefore, the more entities a dialogue contains, the more knowledge it requires. We utilize the average number (K) of entity classes in the dialogue history to identify dialogues that rely on rich knowledge. The average number of entity classes in the SMD and Multi-WOZ 2.1 test sets is 3 and 8, respectively. Through the average number K, we split the test set into two parts. Fig. 5 shows the results of the models on the two parts separately.

We can observe that, for the dialogues that rely on rich knowledge, our method achieves considerable improvements. This demonstrates the effectiveness of our method, which provides a great way to jointly encode the dialogue history and the knowledge base to capture relevant information. In the dialogues that rely on rich knowledge, there are more entities mentioned in the dialogue history that provide more cross-subgraph paths, which contribute to a better understanding of the semantics of the dialogues. On the other hand, through these rich relational paths, each entity gains more graph structure information from the knowledge base according to the current dialogue history.

5.7. Human evaluation

We then answer the RQ7. We provide human evaluations on our model and other baselines. We evaluated 200 responses based on distinct dialogue histories in the SMD test set. We hire two human experts and ask them to judge the quality of the

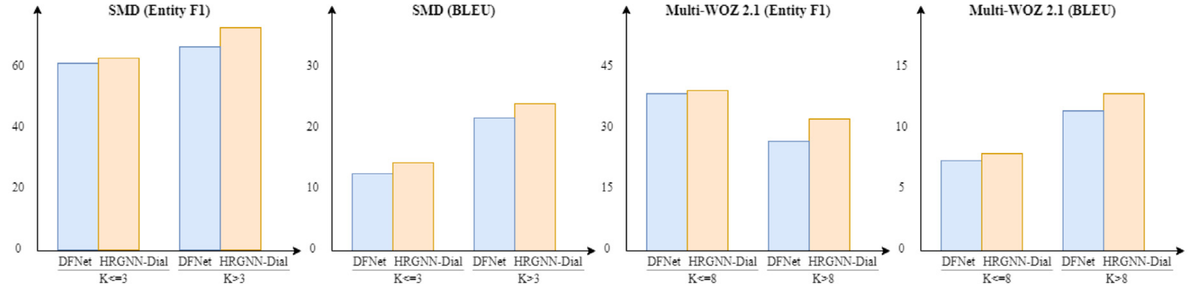


Fig. 5. Test results on dialogues relying on rich knowledge and others. K is the average number of entity classes in the dialogue history.

Table 11
Results of the models that adopt different encoders.

Model	SMD (BLEU)		SMD (Entity F1)		Multi-WOZ 2.1 (BLEU)		Multi-WOZ 2.1 (Entity F1)	
	Test	Δ	Test	Δ	Test	Δ	Test	Δ
HRGNN-AO	16.5	–	**65.2**	–	**10.1**	–	38.0	–
with BiLSTM [54]	16.4	−0.1	64.7	−0.5	10.0	−0.1	**38.1**	+0.1
with Transformer [38]	**16.7**	+0.2	65.1	−0.1	9.7	−0.4	37.8	−0.2

responses according to “correct”, “fluent”, and “human-like” [5] on a scale from 1 to 5. In each judgement, the expert is presented with the dialogue history, the knowledge base, the output of an anonymous system, and the gold response. The evaluation results are shown in Table 10. Our model significantly outperforms GLMP and DFNet on all metrics, which is consistent with the automatic evaluation. The most significant improvement (“correct”) indicates that our model can retrieve accurate entities from KBs to generate system responses that match the user requests.

5.8. Selection of the underlying encoder

Finally, we study the RQ8. The results are shown in Table 11. To select an effective encoder, we evaluate the models that replace the BiGRU encoder with two other encoders, i.e., the Bidirectional Long Short-Term Memory (BiLSTM) [54] and the transformer [38]. On both datasets, the two variants do not achieve significant improvements. We argue that all these encoders are capable of obtaining a good original representation of the dialogue history. Based on the original representation, our heterogeneous relational graph neural networks fuse the knowledge base and the dialogue history, which can capture graph structure information to facilitate response generation. Existing task-oriented dialogue systems (e.g., GLMP, DFNet, and ours) usually represent each entity in the knowledge base as a whole token in the vocabulary. Such vocabulary is of great benefit to these models, as it enables these models to represent each entity as an embedding vector and to copy the complete entity into the response. However, the vocabulary of current pre-trained models (e.g, BERT [55]) cannot represent so many special tokens. Therefore, these task-oriented dialogue systems do not adopt pre-trained encoders.

6. Conclusion

In this paper, we propose heterogeneous relational graph neural networks with an adaptive objective for end-to-end task-oriented dialogues. In our method, we exploit heterogeneous relational graphs to jointly encode the dialogue history and the knowledge base. Our method captures the graph structure information from the knowledge base and the relevant information between the two texts. It is an effective way to incorporate external KBs into the learning framework. In addition, we propose a novel graph encoder, which adopts the shared-private parameterization and the hierarchical attention mechanism to handle overfitting and confusion problems. Besides, we propose an adaptive objective to address the entity imbalance problem via balanced training. The experimental results on the SMD and extended Multi-WOZ 2.1 datasets demonstrate the effectiveness of the proposed method, which achieves state-of-the-art performance. Task-oriented dialogue systems have many real-world applications, such as Apple Siri and Microsoft Cortana. We believe that our work is practical and may inspire many future studies.

CRediT authorship contribution statement

Qingbin Liu: Conceptualization, Methodology, Data curation, Software, Writing - review & editing, Writing - original draft. **Guirong Bai:** Writing - review & editing. **Shizhu He:** Writing - review & editing. **Cao Liu:** Writing - review & editing. **Kang Liu:** Writing - review & editing, Supervision. **Jun Zhao:** Funding acquisition, Supervision.

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.

Acknowledgements

The authors wish to thank the editors and anonymous reviewers for their valuable comments and helpful suggestions which greatly improved the paper quality. The work is supported by the National Key Research and Development Program of China under Grant No. 2020AAA0106400, and the National Natural Science Foundation of China under Grant Nos. 61922085 and 61976211. The work is also supported by the Beijing Academy of Artificial Intelligence under Grant No. BAAI2019QN0301, the Key Research Program of the Chinese Academy of Sciences under Grant No. ZDBS-SSW-JSC006, the independent research project of the National Laboratory of Pattern Recognition, China under Grant, the Youth Innovation Promotion Association CAS, China, and the Meituan-Dianping Group.

References

[1] W. He, Y. Sun, M. Yang, F. Ji, C. Li, R. Xu, Multi-goal multi-agent learning for task-oriented dialogue with bidirectional teacher–student learning, Knowl.-Based Syst. 213 (2021) 106667.

[2] P. Yang, D. Ji, C. Ai, B. Li, AISE: Attending to intent and slots explicitly for better spoken language understanding, Knowl.-Based Syst. 211 (2021) 106537.
[3] B. Zhang, X. Xu, X. Li, Y. Ye, X. Chen, Z. Wang, A memory network based end-to-end personalized task-oriented dialogue generation, Knowl.-Based Syst. 207 (2020) 106398.
[4] S. Young, M. Gašić, B. Thomson, J.D. Williams, POMDP-based statistical spoken dialog systems: A review, Proc. IEEE 101 (5) (2013) 1160–1179.
[5] L. Qin, X. Xu, W. Che, Y. Zhang, T. Liu, Dynamic fusion network for multi-domain end-to-end task-oriented dialog, in: Proceedings of the 58th ACL, 2020, pp. 6344–6354.
[6] M. Nakano, K. Komatani, A framework for building closed-domain chat dialogue systems, Knowl.-Based Syst. 204 (2020) 106212.
[7] Y. Shan, Z. Li, J. Zhang, F. Meng, Y. Feng, C. Niu, J. Zhou, A contextual hierarchical attention network with adaptive objective for dialogue state tracking, in: Proceedings of the 58th ACL, 2020, ppp. 6322–6333.
[8] C. Zhu, M. Zeng, X. Huang, Multi-task learning for natural language generation in task-oriented dialogue, in: Proceedings of the 2019 EMNLP-IJCNLP, 2019, pp. 1261–1266.
[9] M. Eric, L. Krishnan, F. Charette, C.D. Manning, Key-value retrieval networks for task-oriented dialogue, in: Proceedings of the 18th SIGDIAL, 2017, pp. 37–49.
[10] H. Wen, Y. Liu, W. Che, L. Qin, T. Liu, Sequence-to-sequence learning for task-oriented dialogue with dialogue state representation, in: Proceedings of the 27th COLING, 2018, pp. 3781–3792.
[11] Y.-L. Tuan, Y.-N. Chen, H.-y. Lee, DyKgChat: Benchmarking dialogue generation grounding on dynamic knowledge graphs, in: Proceedings of the 2019 EMNLP-IJCNLP, 2019, pp. 1855–1865.
[12] C.-S. Wu, R. Socher, C. Xiong, Global-to-local memory pointer networks for task-oriented dialogue, in: Proceedings of the ICLR, 2019.
[13] A. Madotto, C.-S. Wu, P. Fung, Mem2Seq: Effectively incorporating knowledge bases into end-to-end task-oriented dialog systems, in: Proceedings of the 56th ACL, 2018, pp. 1468–1478.
[14] P. Budzianowski, T.-H. Wen, B.-H. Tseng, I. Casanueva, S. Ultes, O. Ramadan, M. Gašić, MultiWOZ - A large-scale multi-domain Wizard-of-OZ dataset for task-oriented dialogue modelling, in: Proceedings of the 2018 EMNLP, 2018, pp. 5016–5026.
[15] M. Eric, R. Goel, S. Paul, A. Sethi, S. Agarwal, S. Gao, D. Hakkani-Tur, MultiWOZ 2.1: Multi-domain dialogue state corrections and state tracking baselines, 2019, arXiv preprint arXiv:1907.01669.
[16] M. Schlichtkrull, T.N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, M. Welling, Modeling relational data with graph convolutional networks, in: ESWC, 2018, pp. 593–607.
[17] D. Busbridge, D. Sherburn, P. Cavallo, N.Y. Hammerla, Relational graph attention networks, 2019, arXiv preprint arXiv:1904.05811.
[18] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, P.S. Yu, Heterogeneous graph attention network, in: WWW, 2019, pp. 2022–2032.
[19] S. Min, Z. Gao, J. Peng, L. Wang, K. Qin, B. Fang, STGSN — A Spatial–temporal graph neural network framework for time-evolving social networks, Knowl.-Based Syst. 214 (2021) 106746.
[20] C.-S. Wu, S.C. Hoi, R. Socher, C. Xiong, TOD-BERT: Pre-trained natural language understanding for task-oriented dialogue, in: Proceedings of the 2020 EMNLP, 2020, pp. 917–929.
[21] Z. Zhang, L. Liao, M. Huang, X. Zhu, T.-S. Chua, Neural multimodal belief tracker with adaptive attention for dialogue systems, in: WWW, 2019, pp. 2401–2412.
[22] N. Vedula, N. Lipka, P. Maneriker, S. Parthasarathy, Open intent extraction from natural language interactions, in: Proceedings of WWW 2020, 2020, pp. 2009–2020.
[23] S. Gao, Y. Zhang, Z. Ou, Z. Yu, Paraphrase augmented task-oriented dialog generation, in: Proceedings of the 58th ACL, 2020, pp. 639–649.
[24] T.-H. Wen, D. Vandyke, N. Mrkšić, M. Gašić, L.M. Rojas-Barahona, P.-H. Su, S. Ultes, S. Young, A network-based end-to-end trainable task-oriented dialogue system, in: Proceedings of the 15th EACL, 2017, pp. 438–449.
[25] T.-H. Wen, Y. Miao, P. Blunsom, S. Young, Latent intention dialogue models, in: Proceedings of the 34th ICML, 2017, pp. 3732–3741.
[26] A. Neelakantan, S. Yavuz, S. Narang, V. Prasad, B. Goodrich, D. Duckworth, C. Sankar, X. Yan, Neural assistant: Joint action prediction, response generation, and latent knowledge reasoning, 2019, arXiv preprint arXiv:1910.14613.
[27] X. Chen, J. Xu, B. Xu, A working memory model for task-oriented dialog response generation, in: Proceedings of the 57th ACL, 2019, pp. 2687–2693.
[28] S. Sukhbaatar, A. Szlam, J. Weston, R. Fergus, End-to-end memory networks, in: C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, R. Garnett (Eds.), NeurIPS, vol. 28, 2015, pp. 2440–2448.
[29] R. Gangi Reddy, D. Contractor, D. Raghu, S. Joshi, Multi-level memory for task oriented dialogs, in: Proceedings of the 2019 NAACL-HLT, 2019, pp. 3744–3754.
[30] L. Qin, Y. Liu, W. Che, H. Wen, Y. Li, T. Liu, Entity-consistent end-to-end task-oriented dialogue system with KB retriever, in: Proceedings of the 2019 EMNLP-IJCNLP, 2019, pp. 133–142.
[31] W. Lei, X. Jin, M.-Y. Kan, Z. Ren, X. He, D. Yin, Sequicity: Simplifying task-oriented dialogue systems with single sequence-to-sequence architectures, in: Proceedings of the 56th ACL, 2018, pp. 1437–1447.
[32] S. Mehri, T. Srinivasan, M. Eskenazi, Structured fusion networks for dialog, in: Proceedings of the 20th SIGDIAL, 2019, pp. 165–177.
[33] Y. Zhang, Z. Ou, Z. Yu, Task-oriented dialog systems that consider multiple appropriate responses under the same context, in: Proceedings of the AAAI, vol. 34, no. 05, 2020, pp. 9604–9611.
[34] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, 2016, arXiv preprint arXiv:1609.02907.
[35] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, Y. Bengio, Graph attention networks, in: ICLR, 2018.
[36] C. Qi, J. Zhang, H. Jia, Q. Mao, L. Wang, H. Song, Deep face clustering using residual graph convolutional network, Knowl.-Based Syst. 211 (2021) 106561.
[37] Y. Fan, J. Liu, W. Weng, B. Chen, Y. Chen, S. Wu, Multi-label feature selection with constraint regression and adaptive spectral graph, Knowl.-Based Syst. 212 (2021) 106621.
[38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: NeurIPS, 2017, pp. 5998–6008.
[39] S. Zhang, L. Xie, Improving attention mechanism in graph neural networks via cardinality preservation, in; IJCAI: Proceedings of the Conference, vol. 2020, 2020, p. 1395.
[40] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, 2015, arXiv preprint arXiv:1512.03385.
[41] D. Wang, P. Liu, Y. Zheng, X. Qiu, X. Huang, Heterogeneous graph neural networks for extractive document summarization, in: Proceedings of the 58th ACL, 2020, pp. 6209–6219.
[42] H. Hong, H. Guo, Y. Lin, X. Yang, Z. Li, J. Ye, An attention-based graph neural network for heterogeneous structural learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 04, 2020, pp. 4132–4139.
[43] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of ICCV, 2017, pp. 2980–2988.
[44] F. Zhou, C. Shui, M. Abbasi, L.-É. Robitaille, B. Wang, C. Gagné, Task similarity estimation through adversarial multitask neural network, IEEE Trans. Neural Netw. Learn. Syst. (2020).
[45] K.E. Thomas, C.J. König, Knowledge of previous tasks: Task similarity influences bias in task duration predictions, Front. Psychol. 9 (2018) 760.
[46] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, E. Hovy, Hierarchical attention networks for document classification, in: Proceedings of the 2016 NAACL-HLT, 2016, pp. 1480–1489.
[47] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder–decoder for statistical machine translation, in; Proceedings of the 2014 EMNLP, 2014, pp. 1724–1734.
[48] I. Sutskever, O. Vinyals, Q.V. Le, Sequence to sequence learning with neural networks, NeurIPS 27 (2014) 3104–3112.
[49] A. Sordoni, M. Galley, M. Auli, C. Brockett, Y. Ji, M. Mitchell, J.-Y. Nie, J. Gao, B. Dolan, A neural network approach to context-sensitive generation of conversational responses, in: Proceedings of the 2015 NAACL-HLT, 2015, pp. 196–205.
[50] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate:, in 3rd ICLR 2015, 2015.
[51] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.
[52] K. Papineni, S. Roukos, T. Ward, W.-J. Zhu, Bleu: a method for automatic evaluation of machine translation, in; Proceedings of the 40th ACL, 2002, pp. 311–318.
[53] M.L. McHugh, Interrater reliability: the kappa statistic, Biochem. Med. 22 (3) (2012) 276–282.
[54] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.
[55] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2018, arXiv preprint arXiv:1810.04805.