



EAGS: An extracting auxiliary knowledge graph model in multi-turn dialogue generation

Bo Ning¹ · Deji Zhao¹ · Xinyi Liu¹ · Guanyu Li¹

Received: 24 April 2022 / Revised: 25 July 2022 / Accepted: 29 August 2022 /
Published online: 30 September 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

Multi-turn dialogue generation is an essential and challenging subtask of text generation in the question answering system. Existing methods focused on extracting latent topic-level relevance or utilizing relevant external background knowledge. However, they are prone to ignore the fact that relying too much on latent aspects will lose subjective key information. Furthermore, there is not so much relevant external knowledge that can be used for referencing or a graph that has complete entity links. Dependency tree is a special structure that can be extracted from sentences, it covers the explicit key information of sentences. Therefore, in this paper, we proposed the EAGS model, which combines the subjective pivotal information from the explicit dependency tree with sentence implicit semantic information. The EAGS model is a knowledge graph enabled multi-turn dialogue generation model, and it doesn't need extra external knowledge, it can not only extract and build a dependency knowledge graph from existing sentences, but also prompt the node representation, which is shared with Bi-GRU each time step word embedding in node semantic level. We store the specific domain subgraphs built by the EAGS, which can be retrieved as external knowledge graph in the future multi-turn dialogue generation task. We design a multi-task training approach to enhance semantics and structure local feature extraction, and balance with the global features. Finally, we conduct experiments on Ubuntu large-scale English multi-turn dialogue community dataset and English Daily dialogue dataset. Experiment results show that our EAGS model performs well on both automatic evaluation and human evaluation compared with the existing baseline models.

Keywords Natural language processing · Knowledge graph · Question answering systems · Dependency tree · Build knowledge graph · Text generation

This article belongs to the Topical Collection: *Special Issue on Knowledge-Graph-Enabled Methods and Applications for the Future Web*

Guest Editors: Xin Wang, Jeff Pan, Qingpeng Zhang, Yuan-Fang Li

✉ Bo Ning
ningbo@dlmu.edu.cn

Extended author information available on the last page of the article

1 Introduction

Dialogue systems tend to directly solve the practical problems in our daily life, which has been filled with much redundant information. The dialogue system can respond quickly to questions from users based on the knowledge base. There are many applications in our real life, such as personal assistants, E-commerce customer service, and chatbots. There are various classifications of dialogue systems, including single-turn and multi-turn dialogue systems according to the type of the dialogue context. While single-turn conversations are well developed, multi-turn conversations have received increasing attention from researchers in recent years because of their more complex contexts and their relevance to real-life scenarios. In multi-turn dialogues, how to use the contextual information effectively becomes an important task in multi-turn dialogues, as there are often multiple turns of interaction between the user and the dialogue system, and the topic in each turn of context often changes.

Existing approaches for multi-turn dialogue generation could be categorized into two groups: handling complex contexts and integrating additional relevant knowledge. The first group tends to construct different methods to deal with the diverse contexts, which are also the traditional and classic ways to solve the multi-turn dialogue generation problem. Researchers have been attempting to extract important information from complex contexts, and one of the earliest efforts on multi-turn dialogue generation is the HRED model proposed by Serbern [1, 2], in which the model multi-turn dialogue is enhanced by adding some additional encoders to HRED. The HRED model provides many useful ideas for modeling multi-turn conversations, but the neuron networks used in it are based on RNN. Later, the Transformer [3] has quickly replaced the RNN-based neural network with its superior performance and computational speed, and there are many researchers working on multi-turn dialogues based on the transformer architecture nowadays. The ReCoSa [4] model intends to use complex attention mechanisms to obtain important word information in context, and a combination of transformer-based self-attention mechanisms and attention mechanisms are used in the ReCoSa model. After obtaining word level information then information fusion is performed and then the decoder is used to decode the generated sentences with maximum probability. However, using this approach still have risks in losing important information about the current turn, the answer results of this model are often repetitive and the decoder is lacking in directionality. Because this model is inadequate for capturing important information about the current turn of conversation, it leads to a lack of thematic consistency in the final generated answers.

The second group of researchers believes that the current multi-turn contexts are not enough to support the diversity of answers, so they hope to find more relevant knowledge and adopt ingenious methods to integrate into the context. Zhang et al. [5] propose the Short-text Topic-level Attention Relevance with Biterm Topic Model (STAR-BTM) model and integrate implicit topic information. However, implicit information is difficult to pay attention to explicit information, like the user's current turn question. Conditional Historical Generation (CHG) [6] tends to utilize more relevant historical dialogues and the model can see the same previous scenario questions. Zhou et al. [7] propose a Commonsense Conversational Model, which can retrieve relevant knowledge graphs from a knowledge base and then encodes the graphs. Although these methods have achieved good results, but in real life, the application scenario of our dialogue systems are very complex, and there is no auxiliary information to help the model improve the quality of sentences. what's

Table 1 In an example of a multi-turn dialogue on the daily dialogue dataset, we split the multi-turn dialogue into many utterances and current turns

Contexts	Examples
Utterance 1	The prices are given without engagement.
Utterance 2	Good, if you'll excuse me. I'll go over the sheet right now.
Utterance 3	Take your time.
Utterance 4	I can tell you at a glance that your prices are much too high.
Utterance 5	You know that the cost of production has been skyrocketing in recent years.
Current Turn	We only ask that your prices be comparable to others.
Response	Well, we can consider making some concessions in our price.

This is an example in the Daily dialogue dataset. The sentences shown are selected. The sentences in the example use word segmentation and sentence segmentation

more, using the retrieval method will greatly increase the parameters of the model, it will increase the time of model training, and slow down the speed of model referencing.

Dependency tree can analyze the sentence structure and parse the sentence into the form of tree including the relationship between words. This structure tree is used by Shi et al. [8] and has achieved good results. It is a great way to transform the tree structure into a graph structure and use the graph embedding method to handle these tree problem. In the process of constructing knowledge map, entity link is also very necessary. Azzalini et al. [9] use deep learning to capture the semantic properties of data. Utilizing the subject-predicate-object triples to build knowledge attracts more attention. Sikos et al. [10] mention many new methods of constructing knowledge graph in a survey.

In this paper, we propose an extracting auxiliary graph structure model in multi-turn dialogue generation, called EAGS. We believe that the syntactic dependency relationship can replace the external related background knowledge, and the structure contains the explicit information of the sentence. Our EAGS model can combine implicit information and explicit information through semantic and structure extracting. we also store the trained subgraphs, which can be retrieved as external knowledge for multi-turn dialogue generation in the specific domain. We split the contexts in the dataset into multiple sentences, and then parse each sentence into a dependency tree. Because the tree structure is a special graph structure, we employ a Graph convolutional neural networks (GCN) to model the graph features. In this paper, we use two publicly available datasets to validate the effectiveness of our model, the Ubuntu multi-turn dialogue dataset [11] and the Daily Dialogue multi-turn dialogue dataset [12]. The relevant experiments validate the effectiveness of our proposed approach.

Table 1 shows an example of a multi-turn conversation selected from the daily dialogue dataset. We segment the contexts according to the user speaking order, which are Utterance 1, Utterance 2, Utterance 3, Utterance 4, Utterance 5 and Current Turn. We believe the current turn contains more useful information, so we give it a higher attention weight. Our proposed EAGS model consists of several existing modules. Although each module is an existing work that has already been proposed, we are the first to utilize the related techniques to combine knowledge graph information for better results in multi-turn dialogue generation task.

The contributions of this paper are summarized as follows:

- We propose the EAGS model, which integrates the syntactic dependency information as the substitution of external knowledge. Due to the employment of syntactic dependency, our model can combine implicit information and explicit information.
- We use graph embedding methods to model dependency tree, and we propose a cross attention method to combine semantic level attention and structure level attention.
- We build the subgraphs that reached the equilibrium state in each context. These specific domain subgraphs can be retrieved as auxiliary knowledge for multi-turn dialogue generation.
- We design a multi-task training approach to enhance graph local features and semantic local features. Multi-task learning method can prompt local characters and balance with global features.
- We conduct experiments on the Ubuntu large-scale English multi-turn dialogue community dataset and Daily dialogue dataset. The experimental results show that our model performs well on both automatic evaluation and human evaluation compared with the existing baseline models.

2 Related work

There are many application scenarios for multi-turn conversations, intelligent customer service bots in e-commerce, voice assistants, blog posts and replies on social media platforms, etc. This data can help users get a better shopping experience, and Yin et al. [13] use NLP technology for tweets on Twitter to analyze the effectiveness of COVID-19. Despite many existing research works on single turn dialogue generation [14–16], multi-turn dialogue generation has gained increasing attention from both academia and industry in recent years. Existing approaches for multi-turn dialogue generation could be categorized into two groups: handling complex contexts and integrating additional relevant knowledge.

2.1 Handling complex contexts

Multi-turn dialogue generation models are mainly based on the encoder decoder architecture, which is proposed by Sequence to Sequence model [1]. This model first started the task of dialogue generation, because it is different from retrieving the answer index from the knowledge base. It can combine the words in the vocabulary dictionary according to the user's questions to generate a logical answer sentence. The Sequence to Sequence (Seq2Seq) model is widely used in single turn dialogue generation. In real life, multi-turn dialogue has more application scenarios. It is obviously unreasonable to simply concatenate multi-turn dialogue as a single-turn dialogue context. So hierarchical recurrent encoder-decoder architectures (HRED) [1] are proposed to capture context information. HRED combines an additional hierarchical encoder to model a part of the conversation. From then on, researchers began to focus on how to deal with complex contexts representation. Later, Serben propose the VHRED [17] model with hidden variables, based on HRED. This method introduces hidden variables into the intermediate state in the previous HRED to improve the diversity of the generated dialogue. The performance of the two models is similar, and VHRED with hidden variables is more robust. Then Transformer [3] has attracted more and more attention because of its ability to extract natural language

features and amazing computing speed. ReCoSa [4] model based on attention mechanism extract the sentence feature and words feature via long self-attention mechanism. However, the ReCoSa model always generates repeat answers, such as ‘I don’t know’, ‘Me too’, and so on. Hierarchical self-attention network (HSAN) [18] can find the most important words and utterances in contexts simultaneously. They use the hierarchical encoder to update the word and utterance representations with their position information respectively.

2.2 Integrating additional relevant knowledge

The second group of researchers believes that it is not enough to only deal with the contexts, so they hope to introduce more external background knowledge, including implicit topic information, context related knowledge map, context related historical dialogue information, and so on. Topic level implicit information mining has a lot of recent works. Zhang et al. [5] propose the STAR-BTM model, which can find latent topic level information and integrates the topic information in the dialogue generation. Xing et al. [19] design a neural topic segmentation model. They enhance a hierarchical attention Bi-LSTM network to better model context, by adding a topic-related auxiliary task and restricted self-attention. Shuai et al. [20] propose a Topic Enhanced Multi-head Co-Attention model (TMCA) based on hierarchical networks to better capture the interactions between sentences via implicit topic information. CMTE model is designed by Li et al. [21], they tend to represent topics with topically related words. The CMTE model focuses not only on coherence with context, but also brings up new chatting topics. Some researchers hope to introduce the knowledge graph structure with external knowledge. Li et al. [22] propose a Topic-level Knowledge-aware Dialogue Generation model to capture context-aware topic-level knowledge information. They decompose the given knowledge graph into a set of topic-level sub-graphs and integrate graph features into their model. Jiang et al. [23] conduct probabilistic topic modeling from the perspective of data privacy in industry. Wu et al. [24] propose a MHKD-Seq2Seq framework to utilise knowledge from other sources. A data manipulation is proposed by Cao et al. [25], which can introduce explicit personas in generation models. However, these models all need to rely on external knowledge, in real life, the multi-turn dialogue task we have to deal with does not have a well-defined knowledge graph structure, and it is very difficult to build a new knowledge graph. There are some works that integrate retrieval and generative methods. Zhu et al. [26] used adversarial training methods to combine generative sentences with sentences obtained from retrieval to get good results, but this method is based on a single turn of dialogue. And the generative adversarial networks (GAN) is hardly to train. CHG [6] focuses more on the integration of dialogue and historical information. Li et al. [27] propose a novel subspace clustering framework, which can map a non-linearly basic theme data into a latent space.

2.3 Graph neural networks

Graphs are a kind of data structure that models a set of objects (nodes) and their relationships (edges). Recently, research on analyzing graphs with machine learning has been receiving more and more attention. Based on CNNs and graph embedding, Graph convolutional neural networks (GCNs) [28] are proposed to collect aggregate information from graph structure. Ying et al. [29] develop a GCN network algorithm, which integrates efficient random walks and convolutions to generate embeddings of nodes. There are many applications that can be used with GCN. Yao et al. [30] use GCN for text classification and

use this method to learn documents and text embeddings for better classification. Li et al. [31] try to use ontology information to constrain the knowledge representation learning model, called TransO. TransO models incorporate rich ontology information with explicit relations. In the process of training entity embedding on knowledge graph, Zhang et al. [32] propose hyperrelational feature learning network (HRFN) to use meta-learned relation features from the dataset. There's also a lot of work on the Knowledge Graph question and answering. Saxena et al. [33] propose an effective method to deal with multi-hop KGQA through sparse Knowledge graphs. In a community question-answering (CQA) system, Jing et al. [34] propose a knowledge-enhanced attentive answer selection model, using this method can consider the professional knowledge and limits of authority. Jian et al. [35] propose a knowledge-aware dialogue generation model to address the issue of introducing common sense into the open domain dialogue system. Atwood et al. [36] incorporate the h-hop transition probability matrix into the convolution operation. Then graph attention network (GAT) [37] is proposed, which can consider the importance of all neighbors of the current node. GAT uses self-attention to build the graph attention layer. Previous approaches are inherently transductive and do not naturally generalize to unseen nodes. GraphSAGE [38] is presented to leverage node feature information and generate node embeddings for previously unseen data. Next, many researchers have done the practical application of graphs, Yu et al. [39] propose a novel deep learning framework to tackle the time series prediction problem in the traffic domain. There are also similar applications of graph in multi-turn dialogue generation. Zhou et al. [7] propose a conversation generation model, which can attentively read the retrieved knowledge graphs and the knowledge triples within each graph to facilitate better generation through a dynamic graph attention mechanism. Xue et al. [40] focus on the dynamic network embedding and refine the category hierarchy by typical learning models. Cai et al. [41] try to use physical interactions and design an influence diffusion model to take into account both cyber and physical user interactions in an effective and practical way. Zhang et al. [42] propose the TransRHS approach, using relational structures to build a more complete knowledge graph. Liu et al. [43] propose a Knowledge Graph Interactive Visual Query Language KGVQL to improve the understanding of knowledge graphs by end users. Knowledge Tracing (KT) can trace the state of evolutionary mastery for particular knowledge or concept, which can also construct a graph structure. Song et al. [44] propose a Bi-CLKT to address the deep knowledge tracing problems, using this method can obtain discriminative representations of concepts based on graph-level contrastive learning. Song et al. [45] try to establish connections between exercises under cross-concepts and enhance model's interpretability.

3 EAGS model

In this section, we will illustrate our model in detail, which architecture is depicted in Figure 1.

3.1 Multi-task training

In this subsection, we will explain how multi-task works. Multi-task learning is a kind of joint learning. Multiple tasks learn in parallel, and the results affect each other.

In the EAGS model, the Node attention and structure are auxiliary tasks to enhance the representation of node and structure respectively. The semantic representation of nodes can affect the structural representation of dependency graphs, and the relationship aggregation

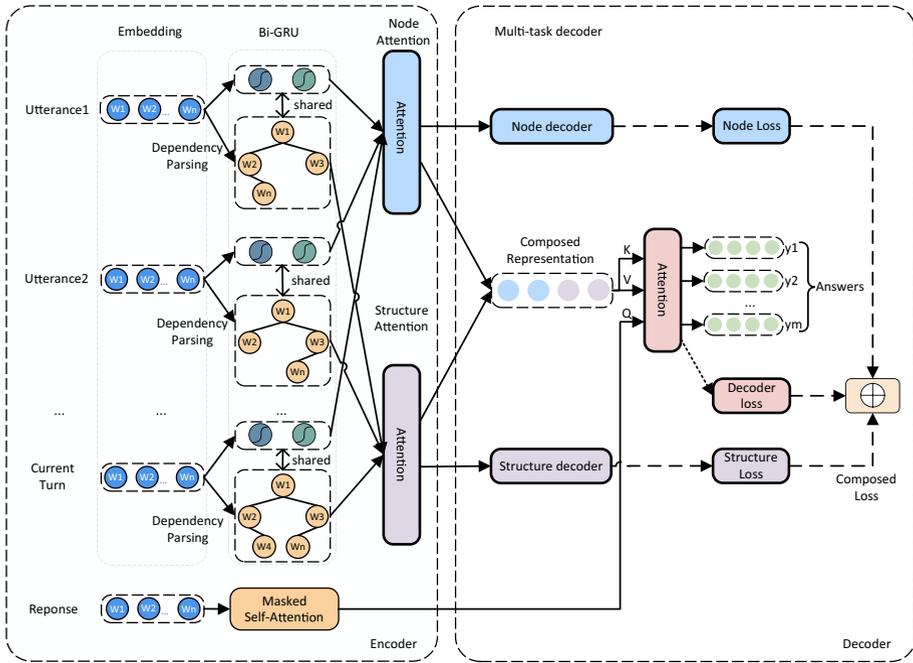


Figure 1 The architecture of EAGS model. The solid lines represent the direction of vector flow, the dotted lines represent the addition of multi-task losses. The blue circles represent the words in a sentence, the box composed with green circles represents the word distribution of vectors

of dependency graphs can also be used as auxiliary information for semantic nodes. Multi-task learning is only used in the training process, and the EAGS model proposed in this paper is somewhat time consuming in training, but many of the computational modules which are time consuming in the model are not used in the actual inference process. There are GCN module, BI-GRU module, encoder module and multi-task decoder module in the EAGS model. These modules compose a large number parameters of EAGS model, but the EAGS model is time consuming during training processing. Once the training of model is done, the time cost during the actual inference process is not significant.

The loss of the Node decoder can be calculated as:

$$Loss_{Node} = P(y_t|C, y_1, \dots, y_{t-1}; \theta) = P(y_t|h^{NR}; \theta) \tag{1}$$

The loss of the Structure decoder can be calculated as:

$$Loss_{Structure} = P(y_t|C, y_1, \dots, y_{t-1}; \theta) = P(y_t|h^{SR}; \theta) \tag{2}$$

The total loss of the EAGS model is the sum of the three losses.

$$Loss = sum(Loss_{Node}, Loss_{Structure}, Loss_{Fusion}) \tag{3}$$

Algorithm 1 shows the calculation process of loss value in multi-task learning. When we get the total loss value, we use Adam optimizer to optimize the whole parameters of the model, which can amplify the effect of local feature extraction of the model and maintain a balance with the global feature. The GCN is jointly optimized with the whole model during

Input: $h^{Node}, h^{Structure}, h^{Res}, Loss_{Fusion}$

Output: $Loss$

- 1: for epoch in number of epochs
 - 2: begin
 - 3: $h^{NR} = Attention(h^{Node}, h^{Res})$
 - 4: $h^{SR} = Attention(h^{Structure}, h^{Res})$
 - 5: $Loss_{Node} = P(y_t | h^{NR}; \theta)$
 - 6: $Loss_{Structure} = P(y_t | h^{SR}; \theta)$
 - 7: $Loss = Loss_{Node} + Loss_{Structure} + Loss_{Fusion}$
 - 8: end;
 - 9: **return** $Loss$
-

Algorithm 1 Multi-task algorithm.

model training, and the node embeddings obtained from BI-GRU are put into the GCN to further fuse structural information. The role of the GCN in the EAGS model is to extract structural information from the syntactic dependency tree, and we use the graph embedding representation to extract structural relationships between different nodes in this paper.

3.2 Hierarchical encoder

In the Multi-turn dialogue generation task, we consider the context contains several sentences, we divide the context into multiple utterances according to the order in which users speak. We believe the most important turn in context is the current turn, which contains questions that users want to ask most in the current turn. Using this method can not only utilise the current turn information as a query to find the most relevant information in previous turns, but also add auxiliary information of user rotation. Furthermore, every separate turn can be parsed by the dependency tree algorithm. The hierarchical architecture we designed is fit for the downstream dependency parsing operations.

Given contexts, which can be divided as follows:

$$Contexts = \{utterance_1, utterance_2, \dots, utterance_n\}, \quad (4)$$

each $utterance_i$ in contexts can be represented as $utterance_i = \{w_1, w_2, \dots, w_n\}$, where w_i represents a word in the utterance.

Different utterances are encoded by various Bi-GRU neural networks in a hierarchical way. Given an utterance $utterance_i$ as input, a standard GRU first encodes each input utterance to a fixed-dimension vector. The GRU is calculated as follows:

$$z_t = \sigma(W_z[h_{t-1}, x_t]) \quad (5)$$

$$r_t = \sigma(W_r[h_{t-1}, x_t]) \quad (6)$$

$$\tilde{h}_t = \tanh(W[h_t * h_{t-1}, x_t]) \quad (7)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \quad (8)$$

where σ is an activation function. We use Bi-directional GRU (Bi-GRU) to get the forward direction latent distribution and backward direction latent direction. In addition, we integrate the positional embedding to the contexts so as to find the importance of different contexts in the sentence level. The TOS embedding refers to the original position embedding [3]. TOS embedding can make the distribution of each turn present a uniform distribution. And using this embedding can make the following attention modules pay more attention to the turn characteristics in the complex contexts. The TOS embedding is calculated as follows:

$$TOS(pos, 2i) = \sin\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right) \quad (9)$$

$$TOS(pos, 2i + 1) = \cos\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right), \quad (10)$$

where d_{model} represents hidden vector dimension in the EAGS model.

The final contexts representation is shown as follows:

$$Contexts = \left\{ \left[\vec{h}_1, \overleftarrow{h}_1 \right] + TOS_1, \dots, \left[\vec{h}_T, \overleftarrow{h}_T \right] + TOS_T \right\}, \quad (11)$$

where TOS_i is the turn of sentence embedding, which represents positional embedding to indicate the order of turns, \vec{h} and \overleftarrow{h} express Bi-directional sentence vector respectively. In the same way, the current turn representation is:

$$CurrentTurn = \left\{ \left[\vec{h}, \overleftarrow{h} \right] \right\} \quad (12)$$

3.3 Graph constructing

In order to transform the utterance into a graph, we employ the syntactic parsing (dependency parsing) algorithm. In the EAGS model, we use Stanford's syntactic analysis method for English. And the syntactic analysis methods can be easily replaced by other methods to transform a specific sentence into a graph structure. The parsing result is a tree structure, which can be represented by graph structure, because tree structure is a special graph structure. The building graph nodes represent the words in the sentence and the edges can express the part of speech relationship between different words, such as nominal subject, passive nominal subject, etc. Table 2 and Figure 2 show the process of parsing sentences and constructing parsing graph. We use utterance 5 and current turn as examples.

From Figure 2, we can see every sentence has been transformed into a graph, which can be expressed by triplets.

3.4 Graph encoder

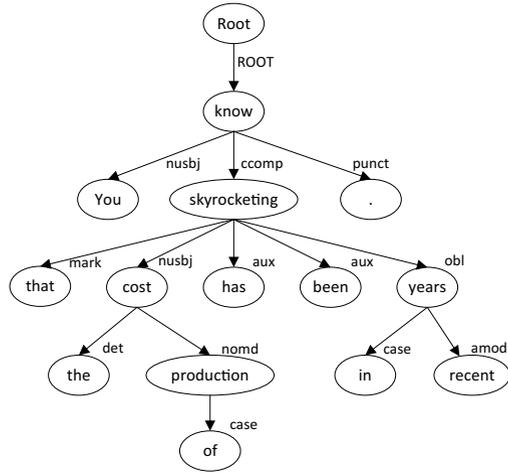
In this subsection, we will explain how to model graph structure. Graph convolutional neural networks (GCN) are widely used as structure encoders for aggregating node information in natural language processing. GCN uses the message passing mechanism to update the current node through its neighbor nodes. Every node will be updated with this mechanism

Table 2 An example in Daily dialogue dataset

Contexts	Examples
Utterance 5	You know that the cost of production has been skyrocketing in recent years.
Current Turn	We only ask that your prices be comparable to others.

Utterance 5:

You know that the cost of production has been skyrocketing in recent years.



Current Turn:

We only ask that your prices be comparable to others.

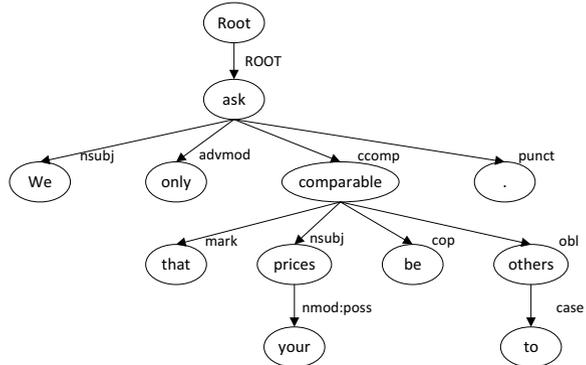


Figure 2 Examples of converting sentences into graphs

in a GCN layer. During the training procedure, there are several layers composed of the complete neural networks. Information interaction between nodes is carried out through multiple layers of the GCN, and eventually the nodes reach a steady state through a messaging passing mechanism. And we consider that the nodes fully absorb the structural information between the syntactic dependency graphs. Figure 3 shows the message passing mechanism, which is employed in the EAGS model. Each GCN node represents a word in the sentence. We use GRU’s every time step outputs as the GCN initial node embeddings instead of random initial in previous work to enhance the representation of the extracted graph on semantic level. Our EAGS training is joint training, so that the representation of each node can not only be updated by neighbor nodes, but also aggregate word semantic vectors. The message passing paradigm between nodes is calculated as :

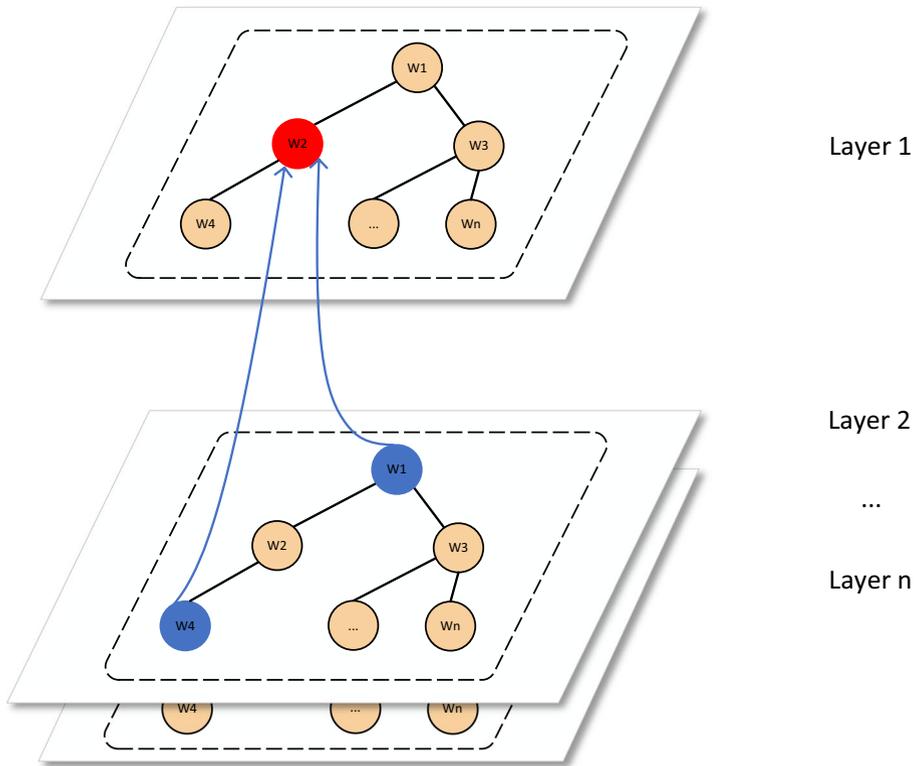


Figure 3 The message passing mechanism of GCN usually has many layers in a GCN network. We use the last layer as the final representation of nodes

$$h^{(l+1)} = \sigma \left(b^{(l)} + \sum_{j \in N(i)} \frac{1}{c_{ji}} h_j^{(l)} W^{(l)} \right), \tag{13}$$

where $h^{(l+1)}$ represents the $l + 1$ layer, $N(i)$ is the set of neighbors of $Node_i$, c_{ji} is the product of the square root of node degrees, $b^{(l)}$ is bias, and σ is an activation function. In this paper, the initialization parameters of GCN are not random, and in order to improve the training speed of the model and make full use of the semantic information of BI-GRU, we use each word embedding representation input to GRU as the initial representation of words in GCN.

The context node representation is:

$$h^{Utterance_i} = Average(w_1, w_2, \dots, w_n), \tag{14}$$

where $Utterance_i$ represents the i^{th} utterance’s sentence vector, w_i represents the word in the $Utterance_i$.

The current turn node representation is:

$$h^{Current} = Average(w_1, w_2, \dots, w_n), \tag{15}$$

where w_i represents the word in the current turn. We use average pooling to get the sentence representation.

3.5 Node level attention

In this subsection, we will express the Node level attention. In the EAGS model, word attention can be seen as node level attention, because the representation of each node corresponds to the word representation in the sentence. To find the critical turn in contexts, we use the attention mechanism [3]. The context attention layer and attention formula are calculated as follows:

$$Attention = softmax\left(\frac{QK^T}{\sqrt{D_k}}\right)V, \quad (16)$$

where Q, K, V represents random matrix respectively and $\sqrt{D_k}$ represents the dimension of the vector. The node level attention can be calculated as follows:

$$h^{Node} = Attention (CurrentTurn, Contexts), \quad (17)$$

where h^{Node} is the Node fusion vector of current turn and contexts.

3.6 Structure level attention

In this subsection, we will express the Structure level attention. In the EAGS model, different utterances in contexts can be extracted as a dependency graph. Different graphs have various structures and relations, which have relevance to the current turn and can be used as auxiliary information to enhance the final fusion vectors. The structure level attention can be calculated as follows:

$$h^{Structure} = Attention (h^{Current}, h^{Utterance_i}), \quad (18)$$

where $h^{Structure}$ is the structure fusion vector of current turn and contexts.

3.7 Decoder

In this subsection, we will present the decoder, jointly training with the hierarchical encoder and graph encoder. Based on the Sequence to Sequence (Seq2Seq) architecture, we used several decoders instead of only one decoder. But the final outputs are only generated by the main decoder, the other two decoders are used to enhance the representation of their own aspects, which are structure representation and node representation. In a multi-task decoder, the sub-decoder has the same structure as the main decoder, consisting of the transformer decoder, but the loss values for the different sub-tasks are calculated separately by the model and finally.

Given an input response: $Response = \{y_1, y_2, \dots, y_m\}$, for each word y_r , we use the mask operator on the response for the training [3] to avoid revealing ground true answers. For each word y_r , we mask $\{y_{t+1}, \dots, y_m\}$ and only see $\{y_1, \dots, y_{t-1}\}$.

$$Response = \{(w_1 + P_1), (w_2 + P_2), \dots, (w_{t-1} + P_{t-1})\}, \quad (19)$$

where P_i represents the positional embedding of words.

In addition, we also used the attention component to feed the matrix of response vectors as queries, keys, and values matrices by using different linear projections. The response vector can be calculated as:

$$h^{Res} = \text{Attention}(\text{Response}), \quad (20)$$

where h^{Res} represents the self-attention outputs vectors.

The node decoder focuses on semantic level enhance, which can be calculated as follow:

$$h^{NR} = \text{Attention}(h^{Node}, h^{Res}), \quad (21)$$

where h^{NR} represents the node hierarchical fusion of contexts and current turn.

The structure decoder focuses on structure level enhance, which can be calculated as follow:

$$h^{SR} = \text{Attention}(h^{Structure}, h^{Res}), \quad (22)$$

where h^{SR} represents the structure hierarchical fusion of contexts and current turn.

The log-likelihood of the corresponding response sequence is:

$$P(Y|C; \theta) = \prod_{t=1}^{T'} P(y_t | C, y_1, \dots, y_{t-1}; \theta), \quad (23)$$

the θ parameter represents the whole model parameters which can be optimized through the different deep neural models.

In order to integrate the node features and structure features, we concatenate the node attention vectors and structure attention vectors.

$$h^{Fusion} = \text{Concat}(h^{Node}, h^{Structure}) \quad (24)$$

After passing the composed attention, which can be seen in the pink box in Figure 1, we can generate words of response through softmax:

$$\text{Loss}_{Fusion} = P(y_t | C, y_1, \dots, y_{t-1}; \theta) = P(y_t | h^{Fusion}; \theta), \quad (25)$$

where $P(y_t)$ is the most likely word in the generated answer sentence and the θ parameter represents the whole model parameters which can be optimized through the different deep neural models.

4 Experiment

4.1 Datasets

We use the Ubuntu community multi-turn dialogue dataset¹ [11] and Daily dialogue dataset² [12] to evaluate the performance of our proposed model. In the multi-turn dialogue dataset, since each turn is distinguished by a special symbol between contexts, this makes it very easy to distinguish each turn in the present paper and incorporate TOS embeddings

¹ <https://github.com/rkadlec/ubuntu-ranking-dataset-creator>

² <https://github.com/Sanghoon94/DailyDialogue-Parser>

to enhance the turn features. We use the official given “__eot__” as a criterion to segment the different turns.

4.2 Baselines

- **Seq2Seq**: Classic Sequence to sequence model with attention mechanism [46].
- **HRED**: Hierarchical Recurrent Encoder-Decoder [1], which add an additional encoder.
- **VHRED**: VHRED is a variant of HRED, the implicit variable information makes the VHRED model more robustness [17].
- **ReCoSa**: Relevant context with self-attention [4] model. Long distance attention and self-attention to capture key words and sentences features.
- **STAR-BTM**: Multi-turn dialogue generation with implicit topic level information [5].
- **CHG**: Utilizing historical dialogue representation to learn a historical dialogue selection model [6].
- **HSAN**: A recent hierarchical self-attention mechanism, which combines the important word features and utterances features in contexts together [18].

4.3 Experiment settings

In order to make a fair comparison between all baseline methods, the hidden layer size is set to 256, the batch size is set to 128, 8 heads attention is used, the Adam optimizer is used, and the learning rate is 0.001. We use Pytorch to run all models on three Tesla T4 GPU.

4.4 Human evaluation

We randomly sampled 200 messages from the Ubuntu test set to conduct the human evaluation as it is extremely time-consuming. We recruit 5 evaluators to judge the response from three aspects [47].

- **Appropriateness**: a response is logical and appropriate to its message.
- **Informativeness**: a response has meaningful information relevant to its message.
- **Grammaticality**: a response is fluent and grammatical.

4.5 Automatic evaluation

We use perplexity[1], BLEU [48] and Dist-1, Dist-2 [49] to evaluate the diversity of our responses, where Dist-k is the number of different k-grams after normalization of the total number of words in the response.

PPL is used in the field of natural language processing to measure the quality of the language model, it is related to the loss of the model. BLEU is to compare the coincidence degree of candidate translation and reference translation. Dist-1 and dist-2 measure the diversity of texts from different angles. Therefore, a good model usually has smaller PPL values and larger BLEU, Dist-1, and Dist-2 values.

4.6 Experiment design

We have done a lot of experiments on both datasets to verify the effectiveness of our EAGS model. Firstly, we compare our proposed multi-turn dialogue generation model EAGS with the existing multi-turn dialogue generation models (as shown in Table 5). In order to verify the necessity and importance of the important modules of our proposed model, we did several groups of experiments.

The EAGS model contains a novel graph structure, it can utilize syntactic features of the sentence itself and it doesn't need external knowledge. In order to ensure the fairness and consistency of the baseline models, we simply added the graph encoder to the other baseline models (as shown in Table 6). The simple fusion method is to directly concatenate the sentence vectors obtained from the average pooling of graph nodes with the other baseline models' sentence vectors. We also employed the several attention mechanisms to extract the most important information from contexts, using this method can give the current turn a higher level of attention weight and maintain more valid information of question. So we removed different modules to check the contribution of each attention module to the model (as shown in Table 7). We didn't use a single decoder to decoder the final answer sentence, we used a multi-task method to help balance the self-promotion of local features and global performance (as shown in Table 8). We also conducted some ablation experiments to verify which module has the greatest improvement on the model (as shown in Table 9).

5 Analysis

The different test results are shown in the tables. We use two evaluation criteria, one is human evaluation, and the other is mainstream evaluation algorithms based on machine translation. In terms of overall performance, our model has achieved good results only through the current contexts information without introducing external knowledge. Using this method is more suitable for the actual situation of lack of data in life.

5.1 Human evaluation

The human evaluation focuses more on areas that are not covered by the automatic assessment. From the human evaluation in Tables 3 and 4, our method has the highest average score in terms of appropriateness, informativeness and grammaticality.

Compared with the most classical Seq2Seq model, our EAGS model has significant improvement in all indicators. Compared with the ReCoSa model, which doesn't integrate any auxiliary information, our EAGS model, our model has significantly improved appropriateness, informativeness, and grammaticality after fusing syntactic structure information. Compared with the model with implicit topic information, the answer sentence generated by the EAGS model with explicit information of syntactic structure and implicit information of semantics are better in terms of human evaluation. This also shows that after the fusion of syntactic structure information, our EAGS model doesn't generate repetitive answers, such as yes, me too, etc. Compared with the model combined with context history information, the EAGS model with each turn of syntactic structure graph generates higher quality answer sentence.

Table 3 Human evaluation results of mean score, proportions of three levels (+2, +1, and 0 represent excellent, good and average respectively) on Ubuntu dataset

Model	Appropriateness				Informativeness				Grammaticality			
	Means	+2	+1	0	Means	+2	+1	0	Means	+2	+1	0
Seq2Seq	0.54	24.9	4.2	70.9	0.39	17.9	3.2	78.9	1.48	71.5	5.0	23.5
ReCoSa	0.69	31.9	5.2	62.9	0.49	22.9	3.2	73.9	1.72	85.7	3.6	10.7
STAR-BTM	0.71	31.5	8.0	60.5	0.54	23.3	7.4	69.3	1.75	84.5	6.0	9.5
CHG	0.81	37.4	6.2	56.4	0.58	26.4	5.2	68.4	1.78	85.6	6.8	7.6
HSAN	0.83	38.6	5.8	55.6	0.66	30.9	4.2	64.9	1.72	81.6	8.8	9.6
EAGS	0.85	39.7	5.6	54.7	0.76	35.9	4.2	59.9	1.85	90.1	4.8	5.1

Table 4 Human evaluation results of mean score, proportions of three levels (+2, +1, and 0 represent excellent, good and average respectively) on Daily dialogue dataset

Model	Appropriateness				Informativeness				Grammaticality			
	Means	+2	+1	0	Means	+2	+1	0	Means	+2	+1	0
Seq2Seq	0.62	29.4	3.2	67.4	0.43	20.6	1.8	77.6	1.51	74.6	1.8	23.6
ReCoSa	0.73	35.5	2.0	62.5	0.51	22.0	7.0	71.0	1.72	83.5	5.0	11.5
STAR-BTM	0.74	34.1	5.8	60.1	0.56	25.3	5.4	69.3	1.75	84.4	6.2	9.4
CHG	0.78	35.1	7.8	57.1	0.64	28.4	7.2	64.4	1.71	80.2	10.6	9.2
HSAN	0.77	35.1	6.8	58.1	0.62	28.7	4.6	66.7	1.75	83.2	8.6	8.2
EAGS	0.89	41.6	5.8	52.6	0.80	38.0	4.0	58.0	1.91	93.1	4.8	2.1

Table 5 Performance of different models on Ubuntu dataset and Daily dialogue dataset

Model	Ubuntu				Daily dialogue			
	PPL	BLEU	Dist-1	Dist-2	PPL	BLEU	Dist-1	Dist-2
Seq2Seq	133.274	0.4813	0.885	0.996	190.147	0.5105	0.683	0.762
HRED	165.112	0.5255	1.021	2.112	178.348	0.5428	0.745	0.984
VHRED	187.054	0.5048	1.125	2.132	175.334	0.5610	0.758	1.032
ReCoSa	99.358	1.1176	1.718	3.768	138.535	1.1828	1.184	2.382
STAR-BTM	111.242	1.2897	1.601	4.525	132.101	1.2105	1.201	2.421
CHG	108.796	1.2814	1.5368	4.438	122.304	1.2236	1.348	2.404
HSAN	101.601	1.3087	1.5971	4.527	120.114	1.2560	1.427	2.488
EAGS	92.884	1.3401	1.6613	4.558	108.561	1.3474	1.454	2.605

5.2 Automatic evaluation

In automatic evaluation experiments, we test our EAGS model on both datasets. From Table 5, we can see our EAGS model has achieved good results in various indicators without introducing external auxiliary knowledge. Compared with the most classical Seq2Seq

Table 6 Comparison of our complete model and simple fusion graph information model on Ubuntu and Daily dialogue dataset

Model	Ubuntu				Daily dialogue			
	PPL	BLEU	Dist-1	Dist-2	PPL	BLEU	Dist-1	Dist-2
Seq2Seq+G	141.285	0.5204	0.941	1.128	199.325	0.5423	0.714	0.851
HRED+G	188.483	0.5417	1.124	2.228	185.159	0.5505	0.751	1.005
VHRED+G	188.265	0.5715	1.230	2.304	183.257	0.5492	0.757	1.087
ReCoSa+G	111.247	1.2231	1.804	3.815	145.232	1.2107	1.203	2.501
STAR-BTM+G	124.753	1.2874	1.719	4.579	149.528	1.2380	1.245	2.527
CHG+G	114.651	1.3018	1.624	4.474	132.470	1.2493	1.441	2.512
HSAN+G	105.790	1.3242	1.632	4.508	124.081	1.2807	1.437	2.538
EAGS	92.884	1.3401	1.6613	4.558	108.561	1.3474	1.454	2.605

Table 7 Attention ablation experiment on Daily dialogue dataset

Model	Node Attention	Structure Attention	PPL↑	BLEU↓	Dist-1↓	Dist-2↓
EAGS	✓	✗	7.104	0.1011	0.015	0.091
EAGS	✗	✓	9.815	0.1208	0.021	0.104
EAGS	✓	✓	0	0	0	0

Table 8 Multi-task ablation experiment on Daily dialogue dataset

Model	Node Decoder	Structure Decoder	PPL↑	BLEU↓	Dist-1↓	Dist-2↓
EAGS	✓	✗	5.662	0.0737	0.011	0.067
EAGS	✗	✓	6.918	0.0924	0.016	0.085
EAGS	✓	✓	0	0	0	0

Table 9 Modules ablation experiment on Daily dialogue dataset

Model	Graph Encoder	Node Encoder	Structure Decoder	Node Decoder	PPL↑	BLEU↓	Dist-1↓	Dist-2↓
EAGS	✗	✗	✗	✗	30.024	0.1517	0.027	0.216
EAGS	✗	✓	✗	✓	22.280	0.1175	0.221	0.158
EAGS	✓	✗	✓	✗	24.715	0.1451	0.021	0.184
EAGS	✓	✓	✓	✓	0	0	0	0

model and HRED model, our EAGS has dramatically improved. Compared with ReCoSa and HSAN models, which use complex operation on contexts, our model performs better after fusing graph information. The dist-1 index of our model on the Ubuntu dataset does not exceed the ReCoSa model, which may be related to the excessive length of multi-turn dialogue sentences in the Ubuntu dataset. On the daily dialogue dataset, the sentence length is not as long as the Ubuntu dataset, so our model achieves very good results. This

may be related to our different splitting methods for contexts and the use of syntactic graph structure information. Compared with STAR-BTM and CHG models, which integrate topic level implicit information and historic background information respectively, the EAGS model with syntactic structure graph is better than them. This should benefit from the combination of explicit information from syntactic parsing of sentences and semantic implicit information of nodes representing words.

In order to ensure the fairness and consistency of the baseline models, we simply added the graph encoder to the other baseline models. From Table 6, we can find that after simply adding graph structure information to the baseline models, the loss of these models increases, but the quality of generating answer sentence is slightly improved. Compared with STAR-BTM and CHG model, we find their performance indicators are improved, we believe that the structure graph information covers explicit information that topic implicit information and historical information don't have.

In the attention ablation experiment in Table 7, we remove the node attention and structure attention respectively. Removing structural attention has the greatest impact on the model. However, no matter which attention module is removed separately, our model performed better than most indicators of the previous baseline models. We believe that structure attention can learn more syntactic relations, such as nominal subject, passive nominal subject, etc. What's more, using distinguished contexts segmentation method according to users' utterance order is also more suitable for the attention mechanism to find the key turns.

In the multi-task ablation experiment in Table 8, we modified the final loss addition method to remove the node decoder loss and structure decoder loss respectively. From Table 8, we can see the performance indicators of the EAGS model had decreased after removing the multi-task module, but the decline of the performance indicators of the model is smaller than that of removing the attention module. We believe that multi-task learning will improve the final performance of the model when there are auxiliary tasks. Our EAGS model can self-correct the structure and semantic aspects of nodes, and the local loss can affect global performance. But using multi-task may increase the parameters of the model and the time to optimize the model.

We removed several important parts of our EAGS model, the graph encoder and decoder are a group, and node encoder and decoder are a group. The graph encoder is responsible for aggregating sentence structures, the node encoder is responsible for optimizing the semantic embedding of each word and each node corresponds to a word in the sentence. From Table 9, we can find that removing node encoder-decoder and graph encoder-decoder have a great impact on the model. After removing the structure module and node module, the downward trend of the model is similar to that of removing the attention module.

In general, we believe that without any external effective knowledge, our EAGS model is more suitable for the actual situation in life. The use of sentence native syntactic dependency information is helpful to the final performance of the model. It is feasible to use GCN to help solve the dependency tree problem, and the use of the multi-task learning method can help the model better aggregate local features.

6 Conclusion and future work

In this paper, we proposed the EAGS model, which can improve the quality of multi-turn dialogue generation without the help of external information. We believed that the syntactic dependency information can replace the external knowledge. We used GCN and multi-task

learning method to fit our model. Furthermore, we stored the subgraphs which are built by the EAGS model, the subgraphs can be used as external knowledge graph information in the future multi-turn dialogue generation task. Experiment results showed the superiority of our proposed EAGS model. In the future, we will continue to mine the user's character information contained in the complex multi-turn contexts and combine it with the dialogue systems.

Acknowledgements The work was supported by the National Natural Science Foundation of China (NSFC, No. 61976032) and the Scientific Research Foundation of Liaoning Provincial Department of Education (No. LJKZ0063). The authors are grateful to the anonymous reviewers for their constructive comments, which have helped improved this work significantly.

Author contribution Bo Ning: Conceptualization, Methodology, Software, Writing, Review and Editing, Visualization, Supervision, Project administration. Deji Zhao: Methodology, Software, Validation, Writing the Original Draft. Xinyi Liu: Formal analysis, Data Curation, Writing the Original Draft. Guanyu Li: Supervision, Project administration, Funding acquisition.

Funding The work was supported by the National Natural Science Foundation of China (NSFC, No. 61976032).

Data availability All data in the experiment is authoritative and available.

Declarations

Human and animal Ethics Not applicable.

Ethics approval and consent to participate This article does not contain any studies involving human participants and/or animals by any of the authors.

Consent for publication All authors have read and agreed to the published version of the manuscript.

Competing interests The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Serban, I., Sordoni, A., Bengio, Y., Courville, A., Pineau, J.: Building end-to-end dialogue systems using generative hierarchical neural network models. In: Proceedings of the AAAI conference on artificial intelligence, vol. 30 (2016)
2. Sordoni, A., Bengio, Y., Vahabi, H., Lioma, C., Grue Simonsen, J., Nie, J.-Y.: A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In: Proceedings of the 24th ACM international on conference on information and knowledge management, pp 553–562 (2015)
3. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems NIPS 2017, pp 5998–6008 (2017)
4. Zhang, H., Lan, Y., Pang, L., Guo, J., Cheng, X.: Recosa: Detecting the relevant contexts with self-attention for multi-turn dialogue generation. In: Proceedings of ACL 2019, Volume 1: Long Papers, pp 3721–3730 (2019)
5. Zhang, H., Lan, Y., Pang, L., Chen, H., Ding, Z., Yin, D.: Modeling topical relevance for multi-turn dialogue generation. In: Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI (2020)
6. Zhang, W., Song, K., Kang, Y., Wang, Z., Sun, C., Liu, X., Li, S., Zhang, M., Si, L.: Multi-turn dialogue generation in e-commerce platform with the context of historical dialogue. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, pp 1981–1990 (2020)
7. Zhou, H., Young, T., Huang, M., Zhao, H., Xu, J., Zhu, X.: Commonsense knowledge aware conversation generation with graph attention. In: IJCAI, pp 4623–4629 (2018)

8. Shi, Z., Huang, M.: A deep sequential model for discourse parsing on multi-party dialogues. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp 7007–7014 (2019)
9. Azzalini, F., Jin, S., Renzi, M., Tanca, L.: Blocking techniques for entity linkage: A semantics-based approach. *Data Sci. Eng.* **6**(1), 20–38 (2021)
10. Sikos, L.F., Philp, D.: Provenance-aware knowledge representation: A survey of data models and contextualized knowledge graphs. *Data Sci. Eng.* **5**(3), 293–316 (2020)
11. Lowe, R., Pow, N., Serban, I., Pineau, J.: The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In: Proceedings of the SIGDIAL 2015 Conference, The 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pp 285–294 (2015)
12. Li, Y., Su, H., Shen, X., Li, W., Cao, Z., Niu, S.: Dailydialog: A manually labelled multi-turn dialogue dataset. In: Proceedings of the Eighth International Joint Conference on Natural Language Processing, IJCNLP 2017, Taipei, Taiwan, November 27 - December 1, 2017 - Volume 1: Long Papers, pp 986–995 (2017)
13. Yin, H., Song, X., Yang, S., Li, J.: Sentiment analysis and topic modeling for covid-19 vaccine discussions. *World Wide WEB* **25**(3), 1067–1083 (2022)
14. Li, J., Monroe, W., Shi, T., Jean, S., Ritter, A., Jurafsky, D.: Adversarial learning for neural dialogue generation. [arXiv:1701.06547](https://arxiv.org/abs/1701.06547) (2017)
15. Mou, L., Song, Y., Yan, R., Li, G., Zhang, L., Jin, Z.: Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation. [arXiv:1607.00970](https://arxiv.org/abs/1607.00970) (2016)
16. Zhang, H., Lan, Y., Guo, J., Xu, J., Cheng, X.: Reinforcing coherence for sequence to sequence model in dialogue generation. In: IJCAI, pp 4567–4573 (2018)
17. Serban, I., Sordoni, A., Lowe, R., Charlin, L., Pineau, J., Courville, A., Bengio, Y.: A hierarchical latent variable encoder-decoder model for generating dialogues. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31 (2017)
18. Kong, Y., Zhang, L., Ma, C., Cao, C.: Hsan: A hierarchical self-attention network for multi-turn dialogue generation. In: ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp 7433–7437. IEEE (2021)
19. Xing, L., Hackinen, B., Carenini, G., Trebbi, F.: Improving context modeling in neural topic segmentation. In: Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing, pp 626–636 (2020)
20. Shuai, P., Wei, Z., Liu, S., Xu, X., Li, L.: Topic enhanced multi-head co-attention: Generating distractors for reading comprehension. In: 2021 International Joint Conference on Neural Networks (IJCNN), pp 1–8. IEEE (2021)
21. Li, W., Ge, F., Cai, Y., Ren, D.: A conversational model for eliciting new chatting topics in open-domain conversation. *Neural Netw.* **144**, 540–552 (2021)
22. Li, J., Huang, Q., Cai, Y., Liu, Y., Fu, M., Li, Q.: Topic-level knowledge sub-graphs for multi-turn dialogue generation. *Knowl.-Based Syst.* **234**, 107499 (2021)
23. Jiang, D., Tong, Y., Song, Y., Wu, X., Zhao, W., Peng, J., Lian, R., Xu, Q., Yang, Q.: Industrial federated topic modeling. *ACM Trans. Intell. Syst. Technol. (TIST)* **12**(1), 1–22 (2021)
24. Wu, S., Wang, M., Li, Y., Zhang, D., Wu, Z.: Improving the applicability of knowledge-enhanced dialogue generation systems by using heterogeneous knowledge from multiple sources. In: Proceedings of the Fifteenth ACM International Conference on WEB Search and Data Mining, pp 1149–1157 (2022)
25. Cao, Y., Bi, W., Fang, M., Shi, S., Tao, D.: A model-agnostic data manipulation method for persona-based dialogue generation. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp 7984–8002 (2022)
26. Zhu, Q., Cui, L., Zhang, W., Wei, F., Liu, T.: Retrieval-enhanced adversarial training for neural response generation. In: Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL, pp 3763–3773 (2019)
27. Li, C., Yang, C., Liu, B., Yuan, Y., Wang, G.: Lrsc: Learning representations for subspace clustering. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp 8340–8348 (2021)
28. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks (2016)
29. Ying, R., He, R., Chen, K., Eksombatchai, P., Hamilton, W.L., Leskovec, J.: Graph convolutional neural networks for WEB-scale recommender systems. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp 974–983 (2018)
30. Yao, L., Mao, C., Luo, Y.: Graph convolutional networks for text classification. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp 7370–7377 (2019)
31. Li, Z., Liu, X., Wang, X., Liu, P., Shen, Y.: Transo: a knowledge-driven representation learning method with ontology information constraints. *World Wide WEB*, 1–23 (2022)

32. Zhang, Y., Wang, W., Chen, W., Xu, J., Liu, A., Zhao, L.: Meta-learning based hyper-relation feature modeling for out-of-knowledge-base embedding. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management, pp 2637–2646 (2021)
33. Saxena, A., Tripathi, A., Talukdar, P.: Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp 4498–4507 (2020)
34. Jing, F., Ren, H., Cheng, W., Wang, X., Zhang, Q.: Knowledge-enhanced attentive learning for answer selection in community question answering systems. *Knowl.-Based Syst.*, 109117 (2022)
35. Wang, J., Liu, J., Bi, W., Liu, X., He, K., Xu, R., Yang, M.: Improving knowledge-aware dialogue generation via knowledge base question answering. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp 9169–9176 (2020)
36. Atwood, J., Towsley, D.: Diffusion-convolutional neural networks. In: Advances in Neural Information Processing Systems, pp 1993–2001 (2016)
37. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y.: Graph attention networks. arXiv: 1710.10903 (2017)
38. Hamilton, W.L., Ying, R., Leskovec, J.: Inductive representation learning on large graphs. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, pp 1025–1035 (2017)
39. Yu, B., Yin, H., Zhu, Z.: Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In: IJCAI (2018)
40. Xue, G., Zhong, M., Li, J., Chen, J., Zhai, C., Kong, R.: Dynamic network embedding survey. *Neuro-computing* **472**, 212–223 (2022)
41. Cai, T., Li, J., Mian, A.S., Sellis, T., Yu, J.X., et al.: Target-aware holistic influence maximization in spatial social networks. *IEEE Transactions on Knowledge and Data Engineering* (2020)
42. Zhang, F., Wang, X., Li, Z., Li, J.: Transrhs: a representation learning method for knowledge graphs with relation hierarchical structure. In: Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, pp 2987–2993 (2021)
43. Liu, P., Wang, X., Fu, Q., Yang, Y., Li, Y.-F., Zhang, Q.: Kgvql: A knowledge graph visual query language with bidirectional transformations. *Knowledge-Based Systems*, 108870 (2022)
44. Song, X., Li, J., Lei, Q., Zhao, W., Chen, Y., Mian, A.: Bi-clkt: Bi-graph contrastive learning based knowledge tracing. *Knowl.-Based Syst.* **241**, 108274 (2022)
45. Song, X., Li, J., Tang, Y., Zhao, T., Chen, Y., Guan, Z.: Jkt: A joint graph convolutional network based deep knowledge tracing. *Inf. Sci.* **580**, 510–523 (2021)
46. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in Neural Information Processing Systems, pp 3104–3112 (2014)
47. Ke, P., Guan, J., Huang, M., Zhu, X.: Generating informative responses with controlled sentence function. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, pp 1499–1508 (2018)
48. Xing, C., Wu, W., Wu, Y., Liu, J., Huang, Y., Zhou, M., Ma, W.-Y.: Topic aware neural response generation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31 (2017)
49. Li, J., Galley, M., Brockett, C., Gao, J., Dolan, B.: A diversity-promoting objective function for neural conversation models. In: NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics, pp 110–119 (2016)

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

Authors and Affiliations

Bo Ning¹ · Deji Zhao¹ · Xinyi Liu¹ · Guanyu Li¹

Deji Zhao
dejizhao@dlmu.edu.cn

Xinyi Liu
lxy869281725@dlmu.edu.cn

Guanyu Li
rabitlee@163.com

¹ School of Information Science and Technology, Dalian Maritime University, Linghai road No.1, Dalian 116026, Liaoning, China