# Unsupervised Learning of Deterministic Dialogue Structure with Edge-Enhanced Graph Auto-Encoder

**Yajing Sun[1,2], Yong Shan[3], Chengguang Tang[4]*, Yue Hu[1,2]*, Yinpei Dai[4], Jing Yu[1,2], Jian Sun[4], Fei Huang[4], Luo Si[4]**

[1]Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2]School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3]Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[4]Alibaba Group, Beijing, China
{sunyajing, huyue, yujing02 }@iie.ac.cn, shanyong18s@ict.ac.cn
{chengguang.tcg, yinpei.dyp, jian.sun, f.huang, luo.si}@alibaba-inc.com

## Abstract

It is important for task-oriented dialogue systems to discover the dialogue structure (i.e. the general dialogue flow) from dialogue corpora automatically. Previous work models dialogue structure by extracting latent states for each utterance first and then calculating the transition probabilities among states. These two-stage methods ignore the contextual information when calculating the probabilities, which makes the transitions between the states ambiguous. This paper proposes a conversational graph (CG) to represent deterministic dialogue structure where nodes and edges represent the utterance and context information respectively. An unsupervised Edge-Enhanced Graph Auto-Encoder (EGAE) architecture is designed to model local-contextual and global-structural information for conversational graph learning. Furthermore, a self-supervised objective is introduced with the response selection task to guide the unsupervised learning of the dialogue structure. Experimental results on several public datasets demonstrate that the novel model outperforms several alternatives in aggregating utterances with similar semantics. The effectiveness of the learned dialogue structured is also verified by more than 5% joint accuracy improvement in the downstream task of low resource dialogue state tracking.

## Introduction

Task-oriented dialogue usually follows a typical dialogue flow, which can be summarized as a dialogue structure. It not only describes internal logical structures of specific dialogue scenarios, but also facilitates several downstream dialogue tasks such as dialogue state tracking (Black et al. 2010; Dai et al. 2020), dialogue summarization (Murray, Renals, and Carletta 2005; Liu, Seneff, and Zue 2010) and dialogue generation (Chen, Xu, and Xu 2019). Conventional dialogue systems usually rely on hand-crafted dialogue structure which is time-consuming and unable to be quickly adapted to new scenarios. It is crucial to discover dialogue structure from existing dialogue corpus automatically.

Recent studies follow an unsupervised manner to discover the dialogue structure from dialogue corpus without labelling efforts. Generally, these methods model dialogue structure by extracting latent states for each utterance first and then calculating the transition probabilities among states. Zhai and Williams (2014) generated topics from conversations and applied Hidden Markov Model (HMM) to model typical dialogue flows upon states composed of topics. Compared with their statistical methods, Shi, Zhao, and Yu (2019) utilized variational recurrent neural network (VRNN) to capture high non-linear dynamics in dialogue and learn discrete variables for each utterance. These two-stage methods focus more on dialogue state representation while ignoring the fact that modeling the transitions using probability without considering the context leads to dialogue transition ambiguous. Figure 1 gives an example of the dialogue structure learned by Shi, Zhao, and Yu (2019).

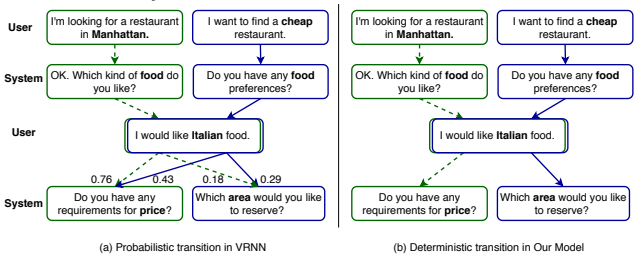


Figure 1: An example of dialogue structures generated by VRNN (Shi, Zhao, and Yu 2019) (a) and our model (b). The solid and dash lines represent different sessions.

The green and blue boxes represent different sessions. After the user replies "*I would like Italian food*", the transition to next state depends on the calculated probability in VRNN (dashed lines). The transition to "*Which area would you like to reserve*" is incorrect for the dash session, and the transition to "*Do you have any requirements for price?*" is incorrect for the solid session. In fact, the dialogue structure should have capability to decide the next transition determinately in the specific context. It's clear that the two-stage method can't obtain a deterministic transition in a specific context.

In this paper, we propose joint-learning of state and transition considering the context information to address

*Corresponding Author

the aforementioned problem, where a novel conversational graph (CG) is used to encode the utterance and context simultaneously as the dialogue structure. The nodes in the CG correspond to the utterances with similar semantics. The edges incorporate the context information to represent the transition relationship which intuitively makes dialogue transition deterministic in a specific context.

While useful to learn a CG with edges containing context information as deterministic dialogue structure, it's nontrivial to learn CG merely from dialogue corpus. Motivated by the cognitive process of human (Dai et al. 2019; Koehn 2017) conversations, the human usually first understand the utterances and then summarize the high-level task-related concept from the context. Afterwards, the connections among concepts are established from the context in these sessions. The two processes are often used to jointly construct dialogue structure subconsciously by humans, which is deterministic based on different contexts.

Therefore, we naturally tackle the challenge by decomposing conversational graph (CG) learning as two modules inspired by human cognition: Semantic Understanding Module (SUM) and Structure Induction Module (SIM). SUM imitates the perception system in the human brain to capture local-contextual information. SIM is analogous to the induction system in the brain, which incorporates specific local-contextual information and global-structural information to induce the conversational graph. In SIM, we intuitively extend Graph Auto-Encoder (GAE) (Kipf and Welling 2016) to a novel Edge-Enhanced Graph Auto-Encoder (EGAE) to unsupervisedly learn conversational graph. Specifically, we initialize CG from the dialogue corpus as prior structural information. EGAE then performs nodes and edges feature aggregation from neighbours by message passing. Moreover we also design a self-supervised objective with the response selection task to guide the unsupervised conversational graph learning. Jointly, SIM and SUM learn the utterance and relationship representations to refine the context-aware conversational graph. As a result, our method obtains a deterministic dialogue structure with CG by learning utterance and relationship information simultaneously.

The main contributions of this paper are three-folds:

- We model the deterministic dialogue structure as a conversational graph with context information. Furthermore, we jointly learn utterances and relationships simultaneously by SUM and SIM, which is motivated by the cognitive process of human.
- To the best of our knowledge, this work is the first attempt to apply an Edge-Enhanced Graph Auto-Encoder (EGAE) architecture to consider local-contextual and global-structural information with end-to-end unsupervised learning. Morever, we introduce the response selection self-supervised task to guide the unsupervised dialogue structure learning.
- Experimental results show that our model outperforms baselines on several task-oriented dialogue datasets. Furthermore, we observe more than 5% improvements in the downstream low-resource dialogue state tracking task, which verifies the effectiveness of the learned structure.

# Related Work

## Unsupervised Dialogue Structure Learning

The challenge of achieving both task completion and human-like response generation for task-oriented dialogue systems is gaining research interest. Previous work tried model end-to-end model with internal (Qiu et al. 2020) or external knowledge (Chen et al. 2020) for high-quality dialogue generation. There are also some previous studies on discovering the latent structure of the conversation. Most of the previous methods utilized the Hidden Markov Model (HMM) to capture the temporal dependencies within human dialogues. Zhai and Williams (2014) decoupled the states and topics and applied hidden markov model (HMM) to model typical dialogue flows upon states which correspond to a mixture of topics. Compared with their statistical methods, Gunasekara et al. (2018) quantized the dialogue space into clusters and created a language model across the clusters, thus allowing for an accurate choice of the next utterance in the conversation. The idea of clustering utterance to create a quantized representation is similar to our method, but we choose the deep learning to capture the semantics representation of utterance and context beyond surface forms of the conversation. Shi, Zhao, and Yu (2019) adopted the variational recurrent neural network (VRNN) to perform variational inference in the model. The VRNN retains the flexibility to model highly non-linear dynamics in dialogues. Different from modeling the dialogue structure as a transition probability, we design a conversational graph to generate a deterministic dialogue structure by modeling the utterance and context simultaneously.

## Graph Auto-Encoder

Recent works have shown that results can often be significantly improved by modeling graph-structured data with end-to-end learning techniques and specifically with graph auto-encoders (Kipf and Welling 2016; Tian et al. 2014; Berg, Kipf, and Welling 2017). Different from Auto-Encoder (Zhao, Xie, and Eskenazi 2019; Wei et al. 2020) used in NLP tasks, Graph Auto-Encoder (GAE) are used to learn meaningful latent embeddings on a social recommendation link prediction task (Berg, Kipf, and Welling 2017) or anomaly detection tasks (Li et al. 2019). Berg, Kipf, and Welling (2017) considered matrix completion for recommendation systems as link prediction on graphs. The model incorporated complementary feature information into the graph to reconstruct the rating links through a bilinear decoder. Gong and Cheng (2019) exploited multi-dimensional edge features and adapted features across the neural network layers. Similarly, we apply GAE to the unsupervised dialogue structure learning with edges containing multi-dimensional context information. It is the first attempt to apply GAE to dialogue task.

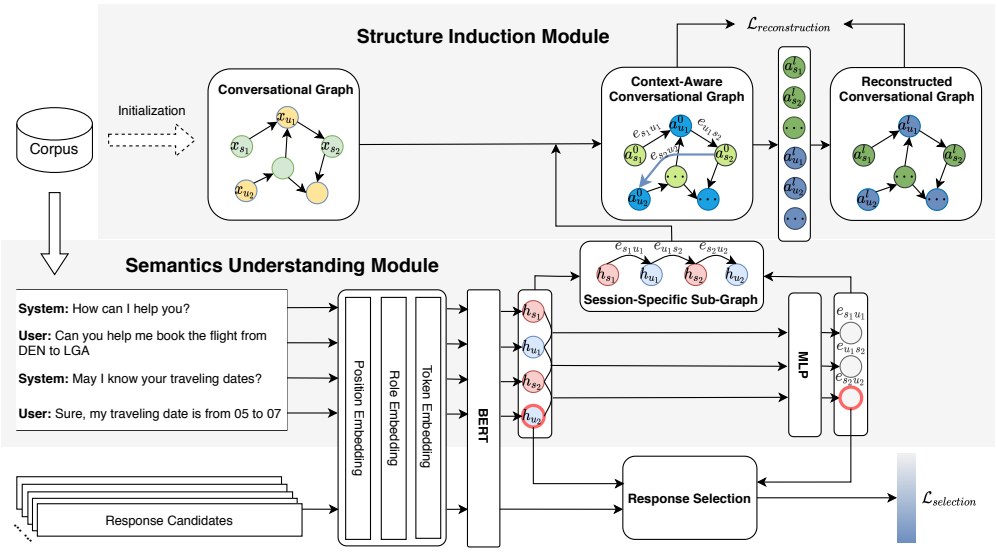


Figure 2: Model Architecture. The model is composed of two modules: Semantics Understanding Module (SUM) and Structure Induction Module (SIM). SUM captures the semantic information from dialogue sessions and builds a session-specific sub-graph. SIM incorporates the sub-graph into the conversational graph and adopts an Edge-Enhanced Graph Auto-Encoder for utterance and relationship learning with end-to-end unsupervised learning.

# Model

## Problem Formulation

Given a dialogue corpus $\mathcal{D}$ where each conversation session consists of $T$ turns of dialogue $\{(u_1, s_1), (u_2, s_2), \cdots, (u_T, s_T)\}$, unsupervised dialogue structure learning extracts the conversational graph with edges containing context information (i.e. the general dialogue flow) from the whole conversation corpus. Here $u$ and $s$ mean user utterance and system response, respectively.

We define a conversational graph (CG) to encode utterance and context simultaneously as dialogue structure. The nodes represent utterances with similar semantics. The edges correspond to the transition with context information. There are $N_u$ and $N_s$ nodes for users and systems, and $N_e$ edges among these nodes. Note that the number of nodes and edges is predefined. We construct CG with a bipartite matrix $M \in \mathbb{R}^{N_u \times N_s}$, where $M_{ij}$ represents the relationship of two nodes. We define $\mathbf{X_u} \in \mathbb{R}^{N_u \times F}$, $\mathbf{X_s} \in \mathbb{R}^{N_s \times F}$ as node features at user side and system side, respectively. Besides, the edge features $\mathbf{X_e} \in \mathbb{R}^{N_e \times F}$ correspond to all possible dialogue context. Motivated by the cognitive process of human, we decompose conversational graph learning into two modules:

(1) **Semantics Understanding Module** imitates the perception system in the human brain. It's responsible for capturing the implicit structure information from each session. The module extracts the semantics information and relationship between the adjacent utterances. The extracted information is used to construct a session-specific sub-graph where nodes correspond to the utterance semantics representation, and edges correspond to the context representations.

(2) **Structure Induction Module** is analogous to logical reasoning system in the brain, which exploits the global structure information for improving semantic understanding capability. The module aims to incorporate context information into the whole conversational graph with unsupervised Edge-Enhanced Graph Auto-Encoder (EGAE). In order to guide the unsupervised structure induction, we also introduce the response selection self-supervised task.

Jointly, the two modules learn the representations of utterances and relationships with an unsupervised manner, which will be used to refine the context-aware conversational graph.

## Semantics Understanding Module

Recently the pre-trained BERT language model (Devlin et al. 2019) shows powerful ability in universal contextual semantics representation. Thus we employ BERT to encode utterances shown in the Figure 2.

For each utterance, the inputs of BERT consist of token embedding, position embedding and role embedding. BERT encodes each utterance into deep contextual representations. Since BERT leverages a special token [CLS] as the whole representation for an utterance, we use a non-linear transformation for [CLS] to obtain the corresponding utterance encoding as follows:

$$[\mathbf{h}_{u_i}, \mathbf{h}_{s_i}] = f(\text{BERT}(u_i, s_i)) \quad (1)$$

where $f$ means the non-linear transformation, $u_i$ and $s_i$ mean the $i$-th user utterance or system utterance, respectively.

For each session $j$, we consider all utterances as a session-specific sub-graph $\mathbf{G}_j$ where the nodes correspond to utterances and the edges correspond to relationships among utterances. Specifically, we establish edges for each two adjacent

utterances and then calculate the relationships between them to obtain the edge features as follows:

$$\mathbf{e}_{u_i,s_i} = \mathbf{W}_e[\mathbf{h}_{u_i}; \mathbf{h}_{s_i}] \quad (2)$$

$$\mathbf{e}_{s_i,u_{i+1}} = \mathbf{W}_e[\mathbf{h}_{s_i}; \mathbf{h}_{u_{i+1}}] \quad (3)$$

where the $\mathbf{W}_e$ is parameters. $G_j$ can capture the relationships between adjacent utterances and provide the implicit structure information to the SIM.

## Structure Induction Module

SIM is responsible for incorporating specific local-contextual information and global-structural information to induce the context-aware conversational graph. The module extends GAE into Edge-enhanced GAE to unsupervised latent nodes and edges representation through differential message passing in the context-aware conversational graph. Since it is quite challenging to learn semantic transitions merely from dialogue corpus, we initialize conversational graph as powerful guidance by aggregating utterances with similar semantics as nodes and establishing edges between nodes containing adjacent utterance in a session. Then EGAE model common structure information with edge information from dialogue corpus with end-to-end unsupervised learning. Different from the previous sequential approaches, the module integrates context into conversational graph considering the local sequential and global structural information. Finally, these latent nodes and edges representations are used to reconstruct the context-aware conversational graph through a score function.

**Graph Initialization** Given a dialogue corpus $\mathcal{D}$, we construct initialized CG with three steps shown in the Algorithm 1: (1) Encoding utterances in the corpus using BERT and splitting them as two parts: user set and system set; (2) Clustering two sets as user/system nodes in the graph using K-Means algorithm respectively; (3) Establishing the edges between nodes if two nodes contain adjacent utterances; (4) Clustering the edges using the K-Means algorithm.

We construct the conversational graph $M$ and label the whole corpus.

**Edge-Enhanced Graph Auto-Encoder** Graph Auto-Encoder (GAE) (Kipf and Welling 2016) are verified effectively in many tasks, and we are the first to apply it in the dialogue structure learning. Different from GAE, our EGAE not only updates the node features but also explicitly adjusts the edge features. The latent node and edge features are used to reconstruct the context-aware conversational graph through a score function, which helps capture adequate context information into the conversational graph.

The context-aware conversational graph $\hat{M}$ is calculated based on the session-specific sub-graph $G_j$ obtained from SUM and the initialized conversational graph $M$. We apply a non-linear layer to fuse the node features of two graphs. Note that we use the edge features in the $G_j$.

The node and edge features are $\mathbf{A}^0$ and $\mathbf{E}^0$ in the $\hat{M}$, respectively. After passing through the $l$ Edge-Enhanced Graph Auto-Encoder layers, $\mathbf{A}^l$ is filtered to produce new

**Algorithm 1:** Conversational Graph Initialization

**Input:** number of user nodes $N_u$, number of system nodes $N_s$, number of edges $N_e$, dialogue corpus $\mathcal{D}$
**Output:** features of user nodes $\mathbf{U_f}$, features of system nodes $\mathbf{S_f}$, conversational graph $M_{ij}$

```
Split D into the user/system utterances sets ;
Use K-Means algorithm to cluster user/system
  utterances into N_u and N_s clusters, respectively ;
Calculate the centroid vectors of user/system clusters
  U_f/S_f ;
for each session in D do
    for each adjacent utterances in session do
        Feed BERT with the concatenated adjacent
          utterances ;
        Insert the [CLS] embedding into the feature
          list of the edge between corresponding
          clusters of adjacent utterances ;
    end
end
for each edge in all edges do
    average the corresponding feature list to obtain
      the edge representations ;
end
Use K-Means algorithm to cluster all edges into N_e
  clusters ;
```

node features $\mathbf{A}^{l+1} \in \mathbb{R}^{N\times F}$. At the same time, edge features are adapted to $\mathbf{E}^{l+1}$, which will be fed to the next layer. The procedure of adapting node and edge features in the layer $l$ is defined as follows:

$$\mathbf{A}^{l+1} = \sigma\left[\alpha^{l+1}\left(\mathbf{A}^l, \mathbf{E}^l\right) g^{l+1}\left(\mathbf{A}^l\right)\right] \quad (4)$$

$$g^{l+1}\left(\mathbf{A}^l\right) = \mathbf{A}^l W^l \quad (5)$$

$$\alpha_{ij}^{l+1} = \text{softmax}(f^l\left(\mathbf{A}_i^l, \mathbf{A}_j^l\right)\mathbf{E}_{ij}^l) \quad (6)$$

$$\mathbf{E}^{l+1} = \alpha^{l+1} \quad (7)$$

where $g$ uses a linear transformation to map the node features from the input space to the output space. $\alpha^{l+1}$ represents the attention coefficients which depends on node features $\mathbf{A}_i^l$, $\mathbf{A}_j^l$ and edge feature $\mathbf{E}_{ij}^l$. Because the edge feature is multi-dimensional, we use separate attention on edge features and combine them by the concatenation operation. For the sake of simplicity, a linear function $f^l$ is regarded as score function.

Furthermore, we employ a score function in decoder to reconstruct conversational graph. Specifically, the decoder produces a probability distribution over all possible dialogue context classes through a linear transformation operation followed by the application of a softmax function:

$$p\left(\hat{M}_{ij}|\mathbf{A}, \mathbf{E}\right) = \text{softmax}(f^l\left(\mathbf{A}_i^l; \mathbf{E}_{ij}^l; \mathbf{A}_j^l\right) \quad (8)$$

| Dataset | CamRest676 | DSTC2 | SGD | | |
|---|---|---|---|---|---|
| | | | Homes | Buses | Flights |
| No. of dialogues | 676 | 1612 | 1273 | 3135 | 3644 |
| No. of slot | 6 | 8 | 2 | 6 | 10 |
| No. of system nodes | 15 | 121 | 26 | 66 | 638 |
| No. of user nodes | 77 | 194 | 95 | 154 | 297 |
| No. of edges | 9 | 13 | 10 | 10 | 15 |

Table 1: Data statistics of DSTC2, CamRest676 and three different domain conversations in the SGD dataset

## Optimization

The EGAE reconstruction loss is calculated as follows:

$$\mathcal{L}_{reconstruction} = -\sum_{i,j}\sum_{r=1}^{R} I[r = M_{ij}] \log p(\hat{M}_{ij}|\mathbf{A}, \mathbf{E})[r] \tag{9}$$

**Response Selection** Response selection helps distinguish whether the response is relevant and consistent with the dialogue context. Meanwhile, as a high quality conversation-related supervised task, it can be regarded as an indicator for extracting high-level semantic information, helping to guide the unsupervised conversational graph learning. Specifically, we randomly sample 19 negative responses candidates for each response from the same set. The goal of the self-supervised task is to learn a scoring model to select a right and proper answer from the candidate answer set. Mathematically:

$$p(r = j) = \frac{e^{f(\mathbf{h}_i^l; \mathbf{e}_{i-1,i}^l)h_j}}{\sum_{k=1}^{K} e^{f(\mathbf{h}i^l; \mathbf{e}_{i-1,i}^l)h_k}} \tag{10}$$

$\mathbf{h}_i^l$ corresponds to the representation of last turn utterance and $\mathbf{e}_{i-1,i}^l$ indicates the last transition representation in the dialogue history. $f$ is non-linear transformation function. The cross-entropy loss as the response selection loss is as follows:

$$\mathcal{L}_{selection} = -\sum_{k=1}^{K} I[r_{selection} = k] \log p(r_{selection} = k) \tag{11}$$

During the training, we optimize our model jointly with $\mathcal{L}_{reconstruction}$ and $\mathcal{L}_{selection}$ as follows:

$$\mathcal{L} = \mathcal{L}_{reconstruction} + \mathcal{L}_{selection} \tag{12}$$

# Experiments

## Setup

**Dataset** We conduct the main experiments on the three public dialogue corpus: DSTC2 (Henderson, Thomson, and Williams 2014), CamRest676 (Rojas-Barahona et al. 2017) and SGD (Rastogi et al. 2019). CamRest676 contains a total of 676 dialogues in this dataset about finding restaurants in Cambridge, UK. Different from the CamRest676, DSTC2 dataset is noisier and more challenging since the bots made mistakes due to speech recognition errors or misinterpretations, which is extracted from real human-bot dialogues. SGD is the largest public task-oriented dialogue corpus, which contains approximately one-third single domain dialogue over 16 domains. We split the single-domain dataset by domain and select home, bus and flight for training. The statistical result is shown in Table 1. Note that the nodes and edges of our conversational graph are predefined based on Table 1, which are equal to the dialogue acts and states annotations in the dataset.

For further verifying the effectiveness of our method, We choose the low-resource dialogue state tracking as our downstream dialogue task, and conduct experiments on WOZ 2.0 dataset (Wen et al. 2017). WOZ 2.0 dataset is a single "restaurant reservation" domain, in which belief trackers estimate three slots (area, food, and price range). Specifically, we sample 10% data from the original training set to construct the low-resource training set. The validation set and the test set are not changed.

**Baseline** We compare our model with the following baseline systems:

- **VRNN** (Shi, Zhao, and Yu 2019) learns a finite state machine of the dialog procedure through a variational autoencoder (VAE) based approach. We replace word2vec with BERT to make comparative experiment fair.

- **BERT & K-Means** employs BERT to obtain the utterance embedding and then clusters them by K-Means (Alsabti, Ranka, and Singh 1997).

**Model Configurations** Our model is implemented with PyTorch [1] (Paszke et al. 2019). We employ the pre-trained BERT model in the SUM module that has 12 layers of 784 hidden units and 12 self-attention heads [2]. The learning rate is 1e-5. In the SIM module, The number of layers and hidden size in EGAE are set to 2 and 768 for the graph convolution encoder, respectively. Adam optimizer (Kingma and Ba 2015) is employed for optimization with learning rate and warmup proportion set to 1e-3 and 0.1. The batch size is set to 64.

**Evaluation Metrics** The evaluation of unsupervised methods has been a challenge. We first use cluster evaluation metrics to evaluate the learned dialogue structure. Then we apply the utterance representation learned from our model

[1] https://pytorch.org/

[2] It is published as *bert-base-uncased* model in a PyTorch version of BERT: https://github.com/huggingface/pytorch-transformers

| Model | CamRest676 | | | DSTC2 | | |
|---|---|---|---|---|---|---|
| | Internal Metrics | | External Metrics | Internal Metrics | | External Metrics |
| | CH ↑ | DB ↓ | FM ↑ | CH ↑ | DB ↓ | FM ↑ |
| VRNN (Shi, Zhao, and Yu 2019) | 6.47 | 4.41 | \ | \ | \ | \ |
| BERT&K-Means | 60.50 | 2.45 | 0.14 | 85.46 | 2.13 | 0.08 |
| Our Model | **639.39** | **1.64** | **0.28** | **668.65** | **1.72** | **0.12** |

Table 2: Experimental results on CamRest676 and DSTC2 datasets. Our model outperforms VRNN (Shi, Zhao, and Yu 2019) and BERT&K-Means baseline systems in both internal and external clustering metrics. "CH", "DB" and "FM" means Calinski-Harabasz Index (Caliński and Harabasz 1974), Davies-Bouldin Index (Davies and Bouldin 1979) and Fowlkes-Mallows scores (Fowlkes and Mallows 1983), respectively. "↑/↓" means the higher/lower the score, the better the clustering performance.

| Domain | Homes | | | Buses | | | Flights | | |
|---|---|---|---|---|---|---|---|---|---|
| Models | CH↑ | DB↓ | FM↑ | CH↑ | DB↓ | FM↑ | CH↑ | DB↓ | FM↑ |
| BERT&K-Means | 98.828 | 2.182 | 0.372 | 72.344 | 2.073 | 0.023 | 53.53 | 2.02 | 0.14 |
| Our Model | **353.606** | **1.303** | **0.386** | **427.678** | **1.438** | **0.222** | **175.18** | **1.45** | **0.156** |

Table 3: Experimental results on home, bus and flight domain in the SGD (Rastogi et al. 2019) datasets. The three domains have different nodes of user and system. Our model outperforms VRNN (Shi, Zhao, and Yu 2019) and BERT&K-Means baseline systems in both internal and external clustering metrics in all domains.

to the low-resource dialogue state tracking task on the WoZ 2.0 dataset.

Clustering evaluation metrics are divided into two types: (1) Internal metrics evaluate the quality of model without any ground-truth, containing Calinski-Harabasz Index (CH) (Caliński and Harabasz 1974) and Davies-Bouldin Index (DB) (Davies and Bouldin 1979), respectively. (2) However, good scores on an internal criterion do not necessarily translate into good effectiveness in an application. We select Fowlkes-Mallows index (FM) (Fowlkes and Mallows 1983) as the external metric which can be used when the ground truth class assignments of the samples are known. Therefore, We construct the ground-truth dialogue structure based on the slot/act annotations of the dataset during test. Specifically, The dialogue acts and turn labels for each turn in the datasets are used to produce node labels such as `inform(price range)` and state labels are regarded as edge labels. Note that in the CamRest676 we calculate the differences between the states of adjacent turns as node labels because there are no turn labels.

For the downstream low-resource dialogue state tracking task, we employ the joint accuracy as the evaluation metric, which is the accuracy of the dialogue state of each turn and a dialogue state is evaluated correctly only if all the values of slots are correctly predicted.

## Main Results

**Cluster Performance** As shown in the Table 2, BERT with K-Means algorithm has significant improvement compared to VRNN model. This result is consistent with the fact that BERT has a sufficient utterance representation capability compared to recurrent neural network (Bengio, Simard, and Frasconi 1994) (RNN). Since there is no ground-truth constructed in Shi, Zhao, and Yu (2019), we do not compare our model with VRNN in the external metrics. Besides, compared to the BERT with K-Means algorithm, the experimental results of our model improve significantly in all metrics. The improvement in the internal metrics proves that our model can aggregate the semantic-close utterance and relationships better. In other words, our model is capable of concentrating more on the high-level concept for each utterance. Moreover, we also observe that the results are better on the CamRest676 dataset than DSTC2, and the reason is that DSTC2 is noisier due to some mistakes made by the bot. Therefore it is also a challenge to filter noisy data in the dataset, especially human-human dialogue. We will explore it in the future work.

| Model | Joint Acc. (%) | Loss |
|---|---|---|
| SUMBT (Lee, Lee, and Kim 2019) | 45.52 | 3.75 |
| SUMBT + Our Model | **51.88** (+6.36) | **3.33** (-0.42) |
| SUMBT + single-sentence LM | 39.01 | 4.43 |
| SUMBT + response-selection LM | 41.01 | 4.08 |
| (w/o init-CG) | 50.83 | 3.36 |
| (w/o self-supervised task) | 47.79 | 3.61 |

Table 4: Ablation study about low-resource (10% training set) dialogue state tracking on WOZ 2.0 dataset.

In addition to DSTC2 and CamRest676 dataset, we also select three different domains in the SGD, which have different number of nodes and edges. Table 3 shows the results. Our model outperforms BERT with K-Means algorithm significantly in all clustering evaluation metrics and all domains, which proves that our utterance representations are more effective due to the elaborate mechanism of modeling the dialogue structure. Meanwhile, we can observe a common phenomenon that performance on both baseline and our model in FM decreases as the number of nodes increases. However, our model is more stable than baseline, which demonstrates the strong generalization capability of our model and BERT with K-Means model has certain limitation to differentiate complex scenarios.

**Application on Downstream Tasks** One of the most important goals for such a structure discovery model is to utilize it to facilitate downstream tasks. Therefore, we conduct experiments to incorporate the structure-augmented utterance encoding into the downstream low-resource dialogue state tracking task. SUMBT (Lee, Lee, and Kim 2019) is a fair and robust baseline compared to our model because it exploits the multi-turn dialogue context and the pre-trained BERT language model. Specifically, we train our model on the full WOZ 2.0 training set and incorporate the utterance representations of our CG into SUMBT (SUMBT+Our Model). The two models subsequently are trained on the dialogue state tracking task with only 10% training data of WOZ 2.0. To show the advantage of the learning representation in the downstream task, we also compare with other existing methods that can also use the global information from the corpus.

- **single-sentence LM** pretraines a domain-adaptive BERT only with masked language model in single sentence.

- **response-selection LM** fine-tunes BERT on sentence selection tasks, another way to capture the partial structure.

As shown in Table 4, the joint accuracy achieves +6.36 improvements and the loss on the test set decreases by 0.42 compared to SUMBT model. The results demonstrate that our method can improve the performance of downstream low-resource dialogue state tracking task in term of both generalization ability and final task accuracy. It indicates that the utterance representations obtained with our unsupervised dialogue structure learning method summarizes local-contextual and global-structural information, which promotes SUMBT model to follow the real-data distribution.

**Ablation Study** We design two ablation experiments: (1) removing the initialization procedure and only reconstructing the session-specific sub-graph. (2) reserving the initialization procedure and removing the self-supervised response selection task. The learned structure-augmented utterance embedding are fed to the downstream low-resource dialogue state tracking task. As shown in the Table 4, the joint accuracy decreases by 1.03 and 4.09 respectively, which demonstrates that unsupervised EGAE and response selection task are beneficial to dialogue structure learning. They promote a higher quality of dialogue structure together, which further boost the performance of downstream task. Besides, joint accuracy decreases by 4.09 after removing the response selection task, which proves that introducing a high quality dialogue-related self-supervised task is useful for guiding the unsupervised dialogue structure learning. Removing the initialization procedure of CG leads to about 1.03 decreases, which demonstrates that the global-structural information is beneficial for dialogue structure learning.

## Case Study

To empirically analyze the quality of the generated dialogue structure, we construct and visualize the conversational graph. The procedure is similar to the initialization of conversational graph other than the utterance and relationship representation is learned from our model instead of

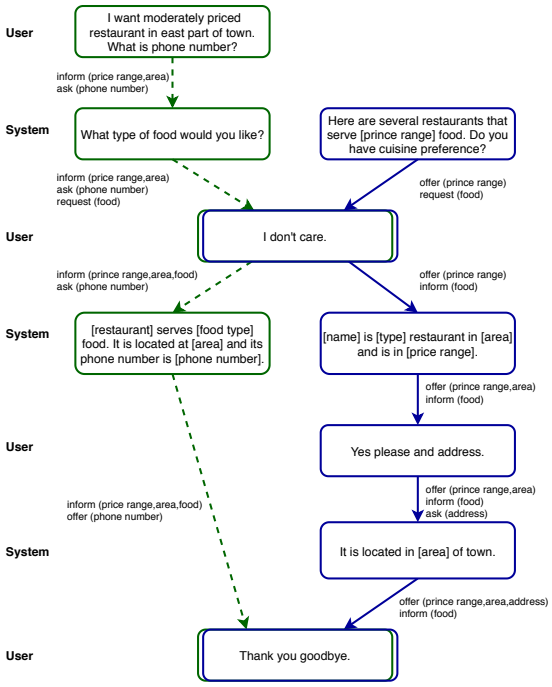


Figure 3: An example of dialogue structures generated by our model.

from BERT. The node is visualized by the Top-1 frequent sentence in each node sentence set. Since sentences in the edges set are long, but the number of edge labels is limited, we apply Key Extractor to extract keywords from Top-10 frequent sentences. Based on the extracted keywords, we manually summarize the edge annotations. Figure 3 shows a part dialogue structure of the bus domain in the SGD dataset. As shown in Figure 3, there are two different dialogue flow in the dialogue structure. Specifically, the node *I don't care* transits different nodes based on the different edges annotations. If the context contains the information that user offers price-range, area and food type information and asks the phone number, the next node will transit to the *[restaurant] serves [food type] they are located at [area] and their phone number is [phone number]*. In contrast, if the context only refers to price-range and food type, the next node will transit *[name] is [type] restaurant in [area] and is in [price range]*. In summary, our model not only can yield a logical flowchart in a completely unsupervised way but also transit between the nodes determinately in the specific context.

## Conclusion

We define dialogue structure as a conversational graph (CG) to obtain a deterministic transition in a specific context and devise the novel Edge-Enhanced Graph Auto-Encoder (EGAE) for graph learning. The experimental results demonstrate that our model performs better in aggregating semantic-close utterances than the baseline. In the future work, we will explore a more effective way to model complex transition relationships in the conversational graph.

## Acknowledgements

We would like to thank all of the anonymous reviewers for their invaluable suggestions and helpful comments. This work was supported by the National Natural Science Foundation of China (Grant No 62006222).

## References

Alsabti, K.; Ranka, S.; and Singh, V. 1997. An efficient k-means clustering algorithm .

Bengio, Y.; Simard, P.; and Frasconi, P. 1994. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks* 5(2): 157–166.

Berg, R. V. D.; Kipf, T. N.; and Welling, M. 2017. Graph Convolutional Matrix Completion .

Black, A. W.; Burger, S.; Langner, B.; Parent, G.; and Eskenazi, M. 2010. Spoken dialog challenge 2010. In *2010 IEEE Spoken Language Technology Workshop*, 448–453. IEEE.

Caliński, T.; and Harabasz, J. 1974. A dendrite method for cluster analysis.

Chen, X.; Meng, F.; Li, P.; Chen, F.; Xu, S.; Xu, B.; and Zhou, J. 2020. Bridging the Gap between Prior and Posterior Knowledge Selection for Knowledge-Grounded Dialogue Generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 3426–3437. Online: Association for Computational Linguistics. doi:10.18653/v1/2020.emnlp-main.275. URL https://www.aclweb.org/anthology/2020.emnlp-main.275.

Chen, X.; Xu, J.; and Xu, B. 2019. A Working Memory Model for Task-oriented Dialog Response Generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2687–2693. Florence, Italy: Association for Computational Linguistics. doi:10.18653/v1/P19-1258. URL https://www.aclweb.org/anthology/P19-1258.

Dai, W.-Z.; Xu, Q.-L.; Yu, Y.; and Zhou, Z.-H. 2019. Bridging Machine Learning and Logical Reasoning by Abductive Learning. In *NeurIPS*.

Dai, Y.; Li, H.; Tang, C.; Li, Y.; Sun, J.; and Zhu, X. 2020. Learning Low-Resource End-To-End Goal-Oriented Dialog for Fast and Reliable System Deployment. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 609–618. Online: Association for Computational Linguistics. doi:10.18653/v1/2020.acl-main.57. URL https://www.aclweb.org/anthology/2020.acl-main.57.

Davies, D. L.; and Bouldin, D. 1979. A Cluster Separation Measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-1: 224–227.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 4171–4186.

Fowlkes, E. B.; and Mallows, C. L. 1983. A Method for Comparing Two Hierarchical Clusterings.

Gong, L.; and Cheng, Q. 2019. Exploiting Edge Features for Graph Neural Networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9203–9211. IEEE.

Gunasekara, R. C.; Nahamoo, D.; Polymenakos, L.; Ganhotra, J.; and Fadnis, K. P. 2018. Quantized-Dialog Language Model for Goal-Oriented Conversational Systems. *ArXiv* abs/1812.10356.

Henderson, M.; Thomson, B.; and Williams, J. D. 2014. The Second Dialog State Tracking Challenge 263–272.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980.

Kipf, T. N.; and Welling, M. 2016. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308* .

Koehn, P. 2017. Cognitive Psychology. In *Encyclopedia of GIS*.

Lee, H.; Lee, J.; and Kim, T.-Y. 2019. SUMBT: Slot-Utterance Matching for Universal and Scalable Belief Tracking. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 5478–5483.

Li, Y.; Huang, X.; Li, J.; Du, M.; and Zou, N. 2019. SpecAE: Spectral AutoEncoder for Anomaly Detection in Attributed Networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2233–2236.

Liu, J.; Seneff, S.; and Zue, V. 2010. Dialogue-Oriented Review Summary Generation for Spoken Dialogue Recommendation Systems. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 2-4, 2010, Los Angeles, California, USA*, 64–72. The Association for Computational Linguistics.

Murray, G.; Renals, S.; and Carletta, J. 2005. Extractive summarization of meeting recordings. In *INTERSPEECH 2005 - Eurospeech, 9th European Conference on Speech Communication and Technology, Lisbon, Portugal, September 4-8, 2005*, 593–596. ISCA.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Köpf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *ArXiv* abs/1912.01703.

Qiu, L.; Zhao, Y.; Shi, W.; Liang, Y.; Shi, F.; Yuan, T.; Yu, Z.; and Zhu, S.-C. 2020. Structured Attention for Unsupervised Dialogue Structure Induction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1889–1899. Online: Association for Computational Linguistics. doi:10.18653/v1/2020.emnlp-main.148. URL https://www.aclweb.org/anthology/2020.emnlp-main.148.

Rastogi, A.; Zang, X.; Sunkara, S.; Gupta, R.; and Khaitan, P. 2019. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset .

Rojas-Barahona, L.; Gaić, M.; Mrksic, N.; Su, P.; Ultes, S.; Wen, T.-H.; Young, S.; and Vandyke, D. 2017. A Network-based End-to-End Trainable Task-oriented Dialogue System. In *EACL*.

Shi, W.; Zhao, T.; and Yu, Z. 2019. Unsupervised Dialog Structure Learning. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 1797–1807.

Tian, F.; Gao, B.; Cui, Q.; Chen, E.; and Liu, T.-Y. 2014. Learning deep representations for graph clustering. In *Aaai*, volume 14, 1293–1299. Citeseer.

Wei, X.; Yu, H.; Hu, Y.; Weng, R.; Xing, L.; and Luo, W. 2020. Uncertainty-Aware Semantic Augmentation for Neural Machine Translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2724–2735. Online: Association for Computational Linguistics. doi:10.18653/v1/2020.emnlp-main.216. URL https://www.aclweb.org/anthology/2020.emnlp-main.216.

Wen, T.-H.; Vandyke, D.; Mrkšić, N.; Gasic, M.; Barahona, L. M. R.; Su, P.-H.; Ultes, S.; and Young, S. 2017. A Network-based End-to-End Trainable Task-oriented Dialogue System. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, 438–449.

Zhai, K.; and Williams, J. D. 2014. Discovering latent structure in task-oriented dialogues. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, 36–46.

Zhao, T.; Xie, K.; and Eskenazi, M. 2019. Rethinking Action Spaces for Reinforcement Learning in End-to-end Dialog Agents with Latent Variable Models. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 1208–1218. Minneapolis, Minnesota: Association for Computational Linguistics. doi:10.18653/v1/N19-1123. URL https://www.aclweb.org/anthology/N19-1123.