Enhancing Dialog Coherence with Event Graph Grounded Content Planning

Jun Xu[1*†], **Zeyang Lei**[2†], **Haifeng Wang**[2], **Zheng-Yu Niu**[2], **Hua Wu**[2] and **Wanxiang Che**[1‡]
[1]Research Center for Social Computing and Information Retrieval,
Harbin Institute of Technology, Harbin, China
[2]Baidu Inc., Beijing, China
{jxu, car}@ir.hit.edu.cn, {leizeyang, wanghaifeng, niuzhengyu, wu_hua}@baidu.com

Abstract

How to generate informative, coherent and sustainable open-domain conversations is a non-trivial task. Previous work on knowledge grounded conversation generation focus on improving dialog informativeness with little attention on dialog coherence. In this paper, to enhance multi-turn dialog coherence, we propose to leverage *event chains* to help determine a sketch of a multi-turn dialog. We first extract event chains from narrative texts and connect them as a graph. We then present a novel event graph grounded Reinforcement Learning (RL) framework. It conducts high-level *response content (simply an event) planning* by learning to walk over the graph, and then produces a response conditioned on the planned content. In particular, we devise a novel multi-policy decision making mechanism to foster a coherent dialog with both appropriate content ordering and high contextual relevance. Experimental results indicate the effectiveness of this framework in terms of dialog coherence and informativeness.

1 Introduction

One of the key goals of AI is to build a machine that can converse with humans by generating informative, coherent and sustainable open-domain conversations. To achieve this goal, end-to-end neural generative models have been studied [Ritter *et al.*, 2011; Shang *et al.*, 2015]. However, these models tend to produce generic responses. To address this issue, some work propose to generate responses by grounding on external knowledge [Dinan *et al.*, 2019; Ghazvininejad *et al.*, 2018; Zhou *et al.*, 2018].

However these knowledge grounded methods tend to generate less coherent dialogs in the setting of multi-turn conversation generation since they focus on improving response informativeness with little attention on multi-turn dialog coherence. In this paper, we make a step towards coherent and informative multi-turn open-domain conversation generation.

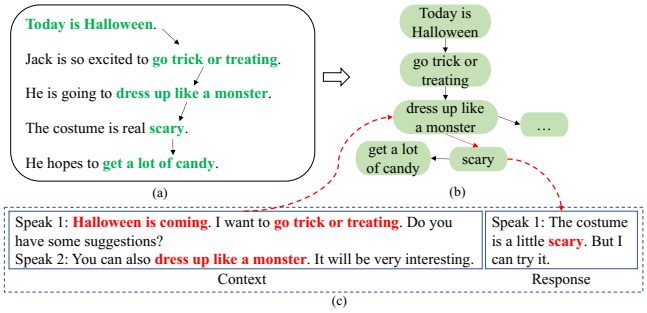

Figure 1: A sample dialog grounded on a narrative event chain. Figure (a) shows a sample event chain (green) extracted from a document. Figure (b) provides a graph with the event chain in (a). Figure (c) shows a coherent multi-turn dialog with appropriate content ordering. The red dotted line refers to the process of selecting an appropriate vertex from the graph to guide dialog generation.

To address this challenge, we propose to leverage the knowledge of narrative event chains to facilitate conversation generation. Narrative event chains are partially ordered sets of events centered around a common protagonist [Chambers and Jurafsky, 2008]. Figure 1 provides a sample event chain extracted from a narrative text. We see that this chain consists of partially ordered events about a single topic. Previous study shows that the use of event chains as background knowledge leads to better coherence judgement of real narrative instances in a narrative cloze task [Chambers and Jurafsky, 2008; Li *et al.*, 2018]. It motivates our study of event chains for conversation generation since the chains might help dialog content ordering, and conditioning on the chains makes it easier to generate coherent dialogs. Figure 1 illustrates a sample dialog conditioned on event chains.

To this end, we present a novel event graph grounded RL framework (**EGRL**). It consists of an event graph, an RL based multi-policy module to conduct explicit high-level response content planning, and a response generator conditioned on the planned content.

First, for event graph construction, we extract event chains from story texts, and connect chains sharing the same events to obtain a *directed* graph. In this graph, vertices represent events (most simply verb phrases), and edges indicate relations between the events.[1] Then we use this graph to facilitate

* This work was done at Baidu.
† Equal contribution.
‡ Corresponding author.

[1]Here, relations refer to temporal order, causal relation, etc.

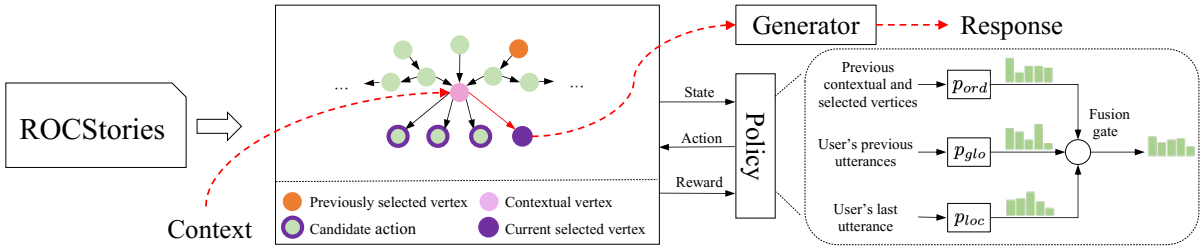

Figure 2: The overall architecture of EGRL. The red dotted line illustrates the data flow in the RL-based policy model.

conversation generation, as shown in Figure 1.

Second, we present a novel event graph grounded RL based multi-policy method to conduct high-level content planning. Given a dialog context, it first links the context to the graph to obtain the contextual vertex. Then it learns to walk along graph edges and identifies an appropriate vertex from one-hop neighbors of the contextual vertex as response content. In this way, our method can utilize event chains directly. Furthermore, to foster a coherent dialog with appropriate content ordering and high contextual relevance, we devise a novel multi-policy decision making mechanism for the RL policy. It consists of three sub-policies: (1) the first sub-policy uses reward signals from a storytelling model [Li *et al.*, 2019] to make the overall structure of multiple responses being consistent with event ordering; (2) the second sub-policy uses reward signals from a topic model [Ramage *et al.*, 2009] to guarantee global response relevance; (3) the third sub-policy uses reward signals from a semantic matching model [Kadlec *et al.*, 2015] to improve local response relevance. Then these sub-policies are fed into a policy-fusion gate for a final decision on content planning. Notice that to avoid "one-sided conversation", we employ two sub-policies (the second one and the third one) to guarantee response content relevance to user message.

Finally, the response generator produces a response conditioned on the planned content and the dialog context.

In summary, this paper makes the following contributions:

- We leverage an event graph to determine a sketch of a multi-turn dialog by RL based content planning, which makes it easier to generate a more coherent dialog.
- To ensure appropriate content ordering and high contextual relevance for content planning, we devise a novel multi-policy decision making mechanism for RL policy.
- Our study indicates that both the event graph and the multi-policy decision making mechanism are crucial to our superior performance in dialog coherence.

2 Our Approach

As shown in Figure 2, the overall architecture of our framework consists of three main parts: an event graph, an RL based multi-policy module and a response generator. Next, we elaborate the details for each of them.

2.1 Event Graph Construction

To obtain the event graph, we first extract event chains from story texts in ROCStories [Mostafazadeh *et al.*, 2016] and then construct the graph based on the extracted chains.

Algorithm 1 Event extraction from each story sentence

Input: A sentence S
Output: A set of events E from S

```
Obtain a dependency parse tree T for S;
Get all the head words HED that are connected to ROOT
    node, and all the leaf nodes in T (denoted as L);
for each leaf node in |L| do
    Extract a phrase consisting of words along the tree
    from HED to current leaf node, denoted as e_i;
    If e_i is a verb phrase, then append it into E;
end for
return E
```

In particular, we first perform dependency parsing for each sentence in story texts to obtain its dependency parse tree.[2] With the obtained dependency parse trees, we extract verb phrases as events for each sentence. The detailed extraction process is presented in Algorithm 1. Then the extracted events are connected in the order they occur in the stories to form event chains.

If two events share no less than 80% words, they will be merged into one event.[3] In this way, we can connect the event chains into a directed graph where vertices are events, and edges represent relations between the events. Formally, the event graph is defined as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ is the set of all vertices and $\mathcal{E}$ is the set of all edges.

2.2 RL Based Multi-Policy Module

To foster a coherent and informative dialog, we propose a novel event graph grounded RL based multi-policy method to conduct high-level response content planning. Given a dialog context (previous two utterances), we first link the context to the graph by retrieving the most related top one vertex as the contextual vertex. Specifically, we utilize string matching to obtain five related vertices[4], and then we retrieve top one vertex from them according to cosine distance in the pre-trained embedding space. We represent a vertex or a dialog context as an average of embeddings of its words. Then the RL-based multi-policy module learns to walk along the edges in the graph and then select an appropriate vertex from one-hop neighbors of the contextual vertex. The selected vertex

[2] https://ai.baidu.com/tech/nlp_basic/dependency_parsing

[3] Here, words refer to noun, verb and adjective words.

[4] We first use the user message at current time step for vertex matching. If no vertices are matched, we will use the system response at previous time step for vertex matching.

will be fed into the response generator to obtain a response. In this way, our method can utilize event ordering information directly. Next we elaborate the main components of RL: state and action, a multi-policy decision making mechanism and reward.

State and action. The current state s_t consists of three parts: $\mathbf{s}_t^v$ to represent the contextual vertex and selected vertices at all previous time steps, $\mathbf{s}_t^u$ to represent the user message at all previous time steps, and $\mathbf{s}_t^l$ to represent current user message. The candidate action set $\mathcal{A}_{s_t} = \{a_i\}_{i=1}^N$ consists of all outgoing one-hop neighbors of the contextual vertex, where N is the number of candidate actions. All states and actions are encoded by Transformers [Vaswani *et al.*, 2017] to obtain their vector representations.

Multi-policy decision making mechanism and rewards. To foster a coherent dialog with appropriate content ordering and high contextual relevance, we first devise a novel multi-policy decision making mechanism that uses different rewards to train three sub-policies. Then three sub-policies are fed into a policy-fusion gate to obtain the final policy and we also design another three rewards to train the final policy.

In particular, to ensure appropriate dialog content ordering, we first employ reward signals from a storytelling model (named as content ordering reward) to train the first sub-policy, named as **content ordering sub-policy**. It helps our policy to make full use of the event ordering information between events in narrative texts for content planning. The sub-policy is formalized as follows:

$$p_{ord}(a_i|\mathbf{s}_t^v) = \frac{\exp({\mathbf{e}_{\mathbf{s}_t^v}}^T \mathbf{e}_{a_i})}{\sum_{j=1}^N \exp({\mathbf{e}_{\mathbf{s}_t^v}}^T \mathbf{e}_{a_j})} \tag{1}$$

where $\mathbf{e}_{\mathbf{s}_t^v}$ and $\mathbf{e}_{a_i}$ denote the vector representations of $\mathbf{s}_t^v$ and the action a_i respectively. The reward value is the prediction probability of a storytelling model [Li *et al.*, 2019].

Second, to improve global relevance (at topic level) of generated responses, we use reward signals from a topic model (named as global relevance reward) to train the second sub-policy, named as **global relevance sub-policy**:

$$p_{glo}(a_i|\mathbf{s}_t^u) = \frac{\exp({\mathbf{e}_{\mathbf{s}_t^u}}^T \mathbf{e}_{a_i})}{\sum_{j=1}^N \exp({\mathbf{e}_{\mathbf{s}_t^u}}^T \mathbf{e}_{a_j})} \tag{2}$$

where $\mathbf{e}_{\mathbf{s}_t^u}$ is the vector representation of $\mathbf{s}_t^u$. For the global relevance reward, we first use a topic model [Ramage *et al.*, 2009] to obtain the topics of the user message at all previous time steps and candidate actions respectively. Then we compute the embedding distances between the topic word embedding of those user message and that of each candidate action as reward values.

Third, to improve local relevance between current response and current user message (single-turn), we employ reward signals from a semantic matching model (named as local relevance reward) to train the third sub-policy, named as **local relevance sub-policy**, presented as follows:

$$p_{loc}(a_i|\mathbf{s}_t^l) = \frac{\exp({\mathbf{e}_{\mathbf{s}_t^l}}^T \mathbf{e}_{a_i})}{\sum_{j=1}^N \exp({\mathbf{e}_{\mathbf{s}_t^l}}^T \mathbf{e}_{a_j})} \tag{3}$$

where $\mathbf{e}_{\mathbf{s}_t^l}$ is the vector representation of $\mathbf{s}_t^l$. We use a BiLSTM-based semantic matching model [Kadlec *et al.*, 2015] to calculate the local relevance. Finally, these three sub-policies are fed into a policy-fusion gate to make a final decision on content planing, which is computed as follows:

$$p_{fin}(a_i|\mathbf{s}_t) = \alpha_1 p_{loc}(a_i|\mathbf{s}_t^v) + \alpha_2 p_{glo}(a_i|\mathbf{s}_t^u) + \alpha_3 p_{loc}(a_i|\mathbf{s}_t^l) \tag{4}$$

Here, α_1, α_2 and α_3 denote the weight coefficients of the three sub-policies. Each of them is computed as follows:

$$\alpha_i = \frac{\sum_{j=1}^3 \mathbf{e}_i \mathbf{e}_j}{\sum_{k=1}^3 \sum_{j=1}^3 \mathbf{e}_k \mathbf{e}_j}, i = 1, 2, 3 \tag{5}$$

where $\mathbf{e}_1$, $\mathbf{e}_2$, $\mathbf{e}_3$ denote $\mathbf{e}_{\mathbf{s}_t^v}$, $\mathbf{e}_{\mathbf{s}_t^u}$, $\mathbf{e}_{\mathbf{s}_t^l}$ respectively.

Rewards

We design three rewards to train the final policy. Specifically, following previous work [Tang *et al.*, 2019; Yao *et al.*, 2018; Zhao *et al.*, 2019], we consider utterance-based rewards shown as follows:

Repetition penalty. The reward is 1 when a generated response shares more than 60% words with one of contextual utterances, otherwise 0.

Moreover, to fully leverage the event graph in policy learning, we employ another two event graph based reward factors:

Global coherence. We calculate the average cosine distance between current selected vertex and all previously selected vertices (or contextual vertex) in TransE based embedding space [Bordes *et al.*, 2013] as global coherence reward. We see that vertices from the same highly connected sub-graph are more likely to constitute coherent dialog and it leads to obtain high global coherence reward.

Sustainability. It is reasonable to give priority to vertices with a large number of neighboring vertices to foster a long-lasting dialog. In particular, we calculate sustainability reward as a PageRank score of the selected vertex. The scores are calculated on the full event graph.

For the final policy, we define its reward as a weighted sum of the above-mentioned three factors with weights whose default values are set as [-0.5, 4, 7000].

At each time step, we sample vertices from the three sub-policies and the final policy by Gumbel-Softmax [Jang *et al.*, 2016] respectively. The vertex sampled by the final policy is utilized to guide response generation.

2.3 Response Generator

The generator produces a response conditioned on the event vertex selected by the policy module and the context. In this work, we use a RNN decoder with a copy mechanism to suit our generation task. In particular, we first build a dataset that suits the settings of our generator by modifying our experimental datasets (Weibo or Twitter Corpus) as follows: (1) sampling an phrase from each gold response as its corresponding event vertex; (2) replacing the sampled phrase with a special symbol, "[event]" in each response. Then we train the response generator with user message and corresponding event vertices as inputs and responses as ground truth.

2.4 Training

To make the RL based training process more stable, we employ the A2C method [Sutton and Barto, 2018] for model optimization. The three sub-policies and the fusion gate are trained simultaneously. Moreover, we only update parameters of the policy module, and parameters of the response generator stay intact during RL training.

3 Experiment

3.1 Datasets

Weibo Corpus. [Shang *et al.*, 2015] The Weibo corpus contains 2.6M message-response pairs for training, 10k pairs for validation and 10k pairs for test.

Twitter Corpus. [Ritter *et al.*, 2011] The corpus contains 1.3M dialogs for training, 10k for validation and 10k for test.

Narrative Event Graph. The ROCStories corpus contains 98,161 five-sentence stories. For the Weibo corpus, we translate the ROCStories corpus into Chinese by Baidu Translate API.[5] Meanwhile, the topic overlap between ROCStories and Weibo (or Twitter) is 98.1% (or 98.5%).[6] Our constructed narrative event graph contains 1,011,547 vertices and 13,668,796 edges. Moreover, to evaluate the quality of the graph, we conduct human evaluation by randomly sampling 500 edges from the graph and then calculate the proportion of edges which are suitable for chatting. The results show that 73.6% of edges are appropriate to dialog.

3.2 Baselines

S2S. It is the widely-used seq2seq model with attention mechanism [Luong *et al.*, 2015].

CCM. It is a commonsense knowledge based conversation model [Zhou *et al.*, 2018], which leverages commonsense knowledge from ConceptNet through two graph attention mechanisms to facilitate informative response generation.[7]

CMR. It is a document augmented neural conversation model [Qin *et al.*, 2019] that jointly models response generation and on-demand machine reading. For fair comparison, we use the ROCStories corpus as the document of CMR.

LaRL. It is a latent variable driven RL based dialog model [Zhao *et al.*, 2019]. We choose the multivariate categorical latent variables as RL actions since it performs the best.

Notice that CCM leverages ConceptNet for generation and CMR uses ROCStories, and we rerun original source codes for CCM [8], CMR [9] and LaRL [10].

We adopt pre-trained Tencent AI Lab Embedding[11] (for Weibo) and Glove[12] (for Twitter) with the size of 200. The vocab size is 50000 and the dimension of all the representations is set to 512. Dropout rate is 0.3. The optimizer adopts Adam and the learning rate is set to 0.002. The discounting weight for reward is 0.95.[13]

3.3 Experimental Settings

Conversation with user simulator. Following the experimental settings in previous work [Li *et al.*, 2016b; Tang *et al.*, 2019], we use a user simulator to play the role of human and let each of the models chat with the same simulator. The user simulator is a pre-trained sequence-to-sequence model with attention mechanism to produce user-side utterances. During the experiments, we use the same user simulator for RL training of our model and baselines. Specifically, given a model to be evaluated, we randomly select an utterance from test set (as the starting position of sessions) to start the conversations with the simulator. Moreover, to avoid infinite conversation, we set maximum number of dialog turns to 8 (i.e. in total, 16 utterances are generated by the simulator and the model) in our experiment. Finally, for each model, we collect 100 model-simulator dialogs to perform multi-turn level evaluation. Meanwhile, for each model, we randomly sample 200 message-response pairs from the model-simulator dialogs for single-turn level evaluation.

Conversation with human. Given a model to be evaluated, we randomly select an utterance from test set for the model to start the conversations with a human turker. Then the human is asked to converse with the model till 8 turns are reached. Finally, we obtain 50 model-human dialogs for multi-turn level evaluation. For single-turn level evaluation, we also randomly sample 200 message-response pairs from model-human dialogs for each model.

3.4 Evaluation Metrics

We define six human evaluation metrics and two automatic metrics. Since the proposed system does not aim at predicting the highest-probability response at each turn, but rather the long-term success of a dialog (e.g., coherence), we do not employ BLEU or perplexity for automatic evaluation [Li *et al.*, 2016b]. (1) Content ordering (**Content-Order.**) for coherence: The metric is used to evaluate whether the ordering of dialog content is appropriate. In practice, we first manually segment a dialog by topics and then conduct evaluation on each sub-topic fragment.[14] A fragment will be rated "1" if the ordering is appropriate, otherwise "0". Finally we compute the average of the scores of all sub-topic fragments over the dialog as content ordering value. (2) Global relevance (**Global-Rele.**) for coherence: Global relevance is used to count the number of incoherent errors within a topic of a dialog. Common incoherence errors in a topic include anaphora errors across utterances and information inconsistency. Similarly,we also perform global relevance evaluation on the above segmented sub-topic fragments. "0" means that there are more than two incoherence errors in a sub-topic fragment,. "1" means that there are one error. "2" means that there are no errors. Finally, we compute the average score

[5] http://fanyi-api.baidu.com/api/trans/product/prodinfo.The translate accurary is 95% by human evaluation.

[6] Here, we use the LDA model [Ramage *et al.*, 2009] to obtain topics of stories and datasets, and the total number of topics is 200.

[7] https://conceptnet.io

[8] https://github.com/tuxchow/ccm

[9] https://github.com/qkaren/converse_reading_cmr

[10] https://github.com/snakeztc/NeuralDialog-LaRL

[11] https://ai.tencent.com/ailab/nlp/embedding.html

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

[13] We optimize these values by grid search.

[14] Each conversation session contains no more than 4 topics.

Methods	Coherence			Informativeness		Overall Quality		
	Content-Order.*	Global-Rele.*	Appr.*	Info.*	Dist-1/2#	Enga.*	Length.#	User-Cons.*
S2S	0.23	0.54	0.40	0.18	0.07/0.21	0.05	2.89	0.22
CCM	0.39	1.03	0.51	0.56	0.12/0.44	0.25	7.71	0.36
CMR	0.14	0.55	0.43	0.47	0.10/0.41	0.18	7.96	0.26
LaRL	0.11	0.48	0.25	0.48	0.12/0.47	0.10	7.93	0.12
EGRL	**0.77**	**1.33**	**0.56**	**0.81**	**0.27/0.69**	**0.75**	**8.00**	**0.72**
EGRL w/o EG	0.31	0.66	0.37	0.79	0.23/0.61	0.18	8.00	0.34
CCM w/ EG	0.46	1.06	0.52	0.59	0.17/0.51	0.34	7.50	0.39
EGRL w/o MP	0.58	1.10	0.53	0.74	0.25/0.66	0.65	8.00	0.66

Table 1: Results for dialogs with user simulator on Weibo corpus. * denotes human evaluation metrics and # denotes automatic metrics.

Methods	Coherence			Informativeness		Overall Quality		
	Content-Order.*	Global-Rele.*	Appr.*	Info.*	Dist-1/2#	Enga.*	Length.#	User-Cons.*
S2S	0.25	0.66	0.45	0.29	0.08/0.25	0.12	4.56	0.22
CCM	0.37	1.04	0.54	0.60	0.17/0.53	0.30	7.72	0.37
CMR	0.20	0.69	0.45	0.52	0.14/0.52	0.22	7.84	0.32
LaRL	0.12	0.54	0.23	0.50	0.14/0.54	0.02	7.98	0.11
EGRL	**0.83**	**1.39**	**0.63**	**0.83**	**0.30/0.77**	**0.80**	**8.00**	**0.76**

Table 2: Results for dialogs with human on Weibo corpus. * denotes human evaluation metrics and # denotes automatic metrics.

of all sub-topic fragments over the dialog as global relevance value. (3) Local relevance or Appropriateness (**Appr.**) for coherence: "0" if a response is inappropriate as an reply, otherwise "1". (4) Informativeness (**Info.**): "0" if a response is a "safe" response, e.g. "I don't know", or it repeats most of the context (more than 80%), otherwise "1". (5) Distinct (**Dist.**): Dist-i calculates the ratio of distinct i-gram in generated responses [Li *et al.*, 2016a]. We use Dist-1 and Dist-2 to measure the diversity of generated responses. (6) Engagement (**Enga.**) for overall quality: This metric measures the overall quality of a dialog. "1" if the dialog has appropriate content ordering, no more than one incoherent errors and responds appropriately to users, otherwise "0". (7) Length-of-dialog (**Length**) for overall quality: Here, we say a dialogue ends when two consecutive utterances from the same agent are highly overlapping [Li *et al.*, 2016b]. (8) User-interests consistency (**User-Cons.**) for overall quality: The metric is used to evaluate if a model can follow a new topic mentioned by a user. A dialog will be rated "1" if the model follows the user's new topic, otherwise "0".

3.5 Experiment Results

We invite three annotators to evaluate each dialog from each model. System identifiers are masked during evaluation.

Results on Weibo Corpus

Conversation with simulators. As shown in Table 1, EGRL significantly outperforms all baselines in terms of all the metrics except for *length-of-dialog* (sign test, p-value $<$ 0.01). It demonstrates that EGRL can effectively foster a more coherent, informative, engaging conversation. In terms of **coherence**, our model outperforms baselines by a large margin in terms of *content ordering*, *global relevance* and *appropriateness*. It indicates that event ordering information and reward signals from a storytelling model can help our model guarantee *content ordering*. And the use of the global and local relevance rewards can help keep responses globally and locally relevant with users. Meanwhile, our model also significantly surpasses baselines in terms of **informativeness** and *Dist-1/2*. In terms of **overall quality** of dialogs, our model obtains the highest scores in terms of *engagement* and *length-of-dialog*. In addition, our model obtains the best *user-interests consistency* result compared with baselines. It indicates that with EGRL, our model avoids one-sided conversation while focusing on dialog coherence. We also observe that S2S tends to generate generic and dull responses, achieving relatively low scores of informativeness and Dist-1/2. CCM amd CMR obtain better informativeness scores than S2S, indicating that incorporating external knowledge into dialog generation can enhance response informativeness. LaRL tends to generate informative but incoherent dialogs, since LaRL's latent variables can not provide sufficient information to accurately guide response generation. The Kappa value for inter-annotator agreement is above 0.4, showing moderate agreement among three annotators.

Conversation with human. As shown in Table 2, EGRL significantly outperforms baselines in terms of all the metrics except for *length-of-dialog* (sign test, p-value $<$ 0.01), which is consistent with the results in Table 1. We observe that scores of most of metrics on Table 2 have been improved in comparison with Table 1. The possible reason is that human can produce higher quality utterances compared with the simulator, e.g., humans rarely fall into a "dead cycle" during conversation, which is helpful for models to produce a longer dialog. Here, the Kappa value is above 0.4.

Ablation Study. We conduct an ablation study in the setting of model-simulator conversation. *First*, to evaluate the contribution of the event graph, we remove the event graph from EGRL, denoted as EGRL w/o EG, where we cannot use

Methods	Coherence			Informativeness		Overall Quality		
	Content-Order.*	Global-Rele.*	Appr.*	Info.*	Dist-1/2#	Enga.*	Length.#	User-Cons.*
S2S	0.17	0.50	0.34	0.20	0.04/0.11	0.03	2.18	0.20
CCM	0.43	0.97	0.50	0.57	0.07/0.23	0.21	7.35	0.32
CMR	0.16	0.64	0.42	0.54	0.07/0.35	0.15	7.98	0.24
LaRL	0.12	0.45	0.24	0.46	0.06/0.21	0.07	7.98	0.12
EGRL	**0.75**	**1.27**	**0.59**	**0.85**	**0.22/0.66**	**0.72**	**8.00**	**0.70**

Table 3: Results for dialogs with user simulator on Twitter corpus. * denotes human evaluation metrics and # denotes automatic metrics.

graph information for action space pruning and reward design. Moreover, we replace ConceptNet with the event graph to augment the CCM model, denoted as CCM w/ EG. As shown in Table 1, the performance of EGRL w/o EG drops dramatically in terms of dialog coherence and informativeness. Furthermore, the event graph can improve the performance of CCM in terms of all the metrics, especially for *content ordering* and *user-interests consistency*. It demonstrates the effectiveness of event graph for appropriate dialog content ordering. *Second*, to verify the effectiveness of multi-policy decision making mechanism, we replace multi-policy mechanism with a single-policy module (merging the inputs of all sub-policies as its input, and all reward items as its reward), denoted as EGRL w/o MP. Results show that the scores of *content ordering* and *appropriateness* drop sharply. It confirms that multi-policy mechanism can enhance coherence.

Results on Twitter Corpus

As shown in Table 3, our model significantly outperforms the baselines in terms of dialog coherence, engagement and informativeness except for *length-of-dialog* (sign test, p-value < 0.01). It is consistent with Table 1.

Case Study

Figure 3 presents an example generated by EGRL. We see that our model can generate more coherent and informative responses. In particular, the ordering of the dialog content by our model is more appropriate, e.g., "tooth decay" $\rightarrow$ "emergency surgery" $\rightarrow$ "recover".

Start: 用薄荷糖和苏打水做喷泉很有趣。(Making a fountain with mentos and soda is fun.)

H: 苏打水还能做喷泉,我还是第一次听说。(I heard it for the first time that fountain can be made with soda.)

B: 然而,这是非常浪费的。(However, it's very wasteful.)

H: 确实很浪费啊,都是食物啊。(Yeah, it does waste food.)

B: 它需要一整卷薄荷糖苏打水。(It requires an entire roll of mentos and a bottle of soda.)

H: 好多啊。(That wastes too much.)

B: 当你把薄荷糖倒进去的时候,苏达就冒出来! (When you dump the mentos in, the soda shoots up!)

H: 我非常喜欢喝苏打水。(I like drinking soda very much.)

B: 我听说有人喝了太多苏打水,牙齿开始腐烂。(I heard someone drank too much soda and it led to tooth decay.)

H: 啊,那也太悲惨了。(How miserable it is!)

B: 他不得不被送进医院做紧急手术。(He had to be taken to the hospital for emergency surgery.)

H: 天啊,后来结果怎么样啊? (Oh my god, what happened later?)

B: 他康复了,几个月后回家了。(He recovered and went home in a few months.)

H: 能康复就是最好的结果。(His recovery is the best ending.)

Figure 3: Case Study. H and B denote human and the bot respectively. The dot line denotes topic segmentation.

4 Related Work

Knowledge-grounded Conversation Generation. There are growing interests in leveraging external knowledge for generation of more informative responses [Ghazvininejad *et al.*, 2018; Moghe *et al.*, 2018; Zhou *et al.*, 2018; Bao *et al.*, 2019; Moon *et al.*, 2019; Liu *et al.*, 2019; Dinan *et al.*, 2019; Xu *et al.*, 2020]. Different from those work, we put more efforts on *dialog coherence* in the setting of multi-turn dialogs. In particular, we care about the ordering of selected knowledge, which is less studied in previous work.

RL based Models for Conversation Generation. Previous work adopt RL based frameworks to learn dialog strategies merely from dialog corpora, which fall into two categories: (1) word-level methods with words as RL actions [Li *et al.*, 2016b; Zhang *et al.*, 2018]; (2) utterance-level methods with high-level utterance representations as RL actions, e.g., latent variables or keywords [Zhao *et al.*, 2019; Yao *et al.*, 2018]. In this work, we investigate how to leverage *external knowledge* to explicitly plot a dialog by RL based content planning.

5 Conclusion

In this paper, we present an event graph grounded RL framework (**EGRL**) to demonstrate how the knowledge of event chains can help plan a multi-turn open-domain dialog. Experimental results on two datasets show that EGRL can generate a more coherent dialog with appropriate content ordering when compared with baselines.

Integrating text knowledge directly for modeling multi-turn dialog logic paves the way for developing models in low-resource domains that lack sufficient dialog corpus. In the future, we will adapt EGRL for low-resource domains.

Acknowledgments

We thank the anonymous reviewers for their helpful comments and suggestions. This work is supported by the National Key Research and Development Project of China (No.2018AAA0101900) and the National Natural Science Foundation of China (NSFC) via grant 61976072.

References

[Bao *et al.*, 2019] Siqi Bao, Huang He, Fan Wang, Rongzhong Lian, and Hua Wu. Know more about each other: Evolving dialogue strategy via compound assessment. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5382–5391, 2019.

[Bordes *et al.*, 2013] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *NIPS*, 2013.

[Chambers and Jurafsky, 2008] Nathanael Chambers and Dan Jurafsky. Unsupervised learning of narrative event chains. In *ACL*, 2008.

[Dinan *et al.*, 2019] Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. Wizard of wikipedia: Knowledge-powered conversational agents. In *ICLR*, 2019.

[Ghazvininejad *et al.*, 2018] Marjan Ghazvininejad, Chris Brockett, Ming-Wei Chang, Bill Dolan, Jianfeng Gao, Wen tau Yih, and Michel Galley. A knowledge-grounded neural conversation model. In *AAAI*, 2018.

[Jang *et al.*, 2016] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[Kadlec *et al.*, 2015] Rudolf Kadlec, Martin Schmid, and Jan Kleindienst. Improved deep learning baselines for ubuntu corpus dialogs. *arXiv preprint arXiv:1510.03753*, 2015.

[Li *et al.*, 2016a] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. In *NAACL-HLT*, 2016.

[Li *et al.*, 2016b] Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. Deep reinforcement learning for dialogue generation. In *EMNLP*, 2016.

[Li *et al.*, 2018] Zhongyang Li, Xiao Ding, and Ting Liu. Constructing narrative event evolutionary graph for script event prediction. 2018.

[Li *et al.*, 2019] Zhongyang Li, Xiao Ding, and Ting Liu. Story ending prediction by transferable bert. In *IJCAI*, 2019.

[Liu *et al.*, 2019] Zhibin Liu, Zheng-Yu Niu, Hua Wu, and Haifeng Wang. Knowledge aware conversation generation with explainable reasoning over augmented graphs. In *EMNLP-IJCNLP*, 2019.

[Luong *et al.*, 2015] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. In *EMNLP*, 2015.

[Moghe *et al.*, 2018] Nikita Moghe, Siddhartha Arora, Suman Banerjee, and Mitesh M. Khapra. Towards exploiting background knowledge for building conversation systems. In *EMNLP*, 2018.

[Moon *et al.*, 2019] Seungwhan Moon, Pararth Shah, Anuj Kumar, and Rajen Subba. Opendialkg: Explainable conversational reasoning with attention-based walks over knowledge graphs. In *ACL*, 2019.

[Mostafazadeh *et al.*, 2016] Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *NAACL*, 2016.

[Qin *et al.*, 2019] Lianhui Qin, Michel Galley, Chris Brockett, Xiaodong Liu, Xiang Gao, Bill Dolan, Yejin Choi, and Jianfeng Gao. Conversing by reading: Contentful neural conversation with on-demand machine reading. In *ACL*, 2019.

[Ramage *et al.*, 2009] Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D Manning. Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. In *ACL*, 2009.

[Ritter *et al.*, 2011] Alan Ritter, Colin Cherry, and William B Dolan. Data-driven response generation in social media. In *EMNLP*, 2011.

[Shang *et al.*, 2015] Lifeng Shang, Zhengdong Lu, and Hang Li. Neural responding machine for short-text conversation. In *ACL*, 2015.

[Sutton and Barto, 2018] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT Press, 2018.

[Tang *et al.*, 2019] Jianheng Tang, Tiancheng Zhao, Chengyan Xiong, Xiaodan Liang, Eric P Xing, and Zhiting Hu. Target-guided open-domain conversation. In *ACL*, 2019.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017.

[Xu *et al.*, 2020] Jun Xu, Haifeng Wang, Zhengyu Niu, Hua Wu, and Wanxiang Che. Knowledge graph grounded goal planning for open-domain conversation generation. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020.

[Yao *et al.*, 2018] Lili Yao, Ruijian Xu, Chao Li, Dongyan Zhao, and Rui Yan. Chat more if you like: Dynamic cue words planning to flow longer conversations. *arXiv preprint arXiv:1811.07631*, 2018.

[Zhang *et al.*, 2018] Wei-Nan Zhang, Lingzhi Li, Dongyan Cao, and Ting Liu. Exploring implicit feedback for open domain conversation generation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[Zhao *et al.*, 2019] Tiancheng Zhao, Kaige Xie, and Maxine Eskenazi. Rethinking action spaces for reinforcement learning in end-to-end dialog agents with latent variable models. In *NAACL-HLT*, 2019.

[Zhou *et al.*, 2018] Hao Zhou, Tom Young, Minlie Huang, Haizhou Zhao, Jingfang Xu, and Xiaoyan Zhu. Commonsense knowledge aware conversation generation with graph attention. In *IJCAI*, 2018.