DlGR-KB: Dual-level Graph Reasoning with Key Block Decoupling for Multi-Party Dialogue Reading Comprehension

Rui Cao[1], Xiabing Zhou[1]*, Min Zhang[2], and Guodong Zhou[1]

[1] School of Computer Science and Technology, Soochow University, Suzhou, China
rcaocaorui@stu.suda.edu.cn,{zhouxiabing,gdzhou}@suda.edu.cn
[2] Harbin Institute of Technology, Harbin, China
zhangmin2021@hit.edu.cn

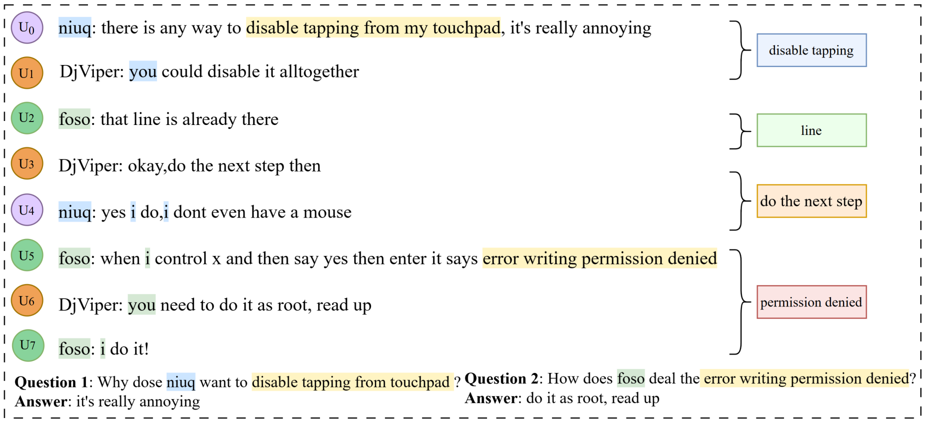
Abstract. Multi-party Dialogue Reading Comprehension (MDRC) is a task focused on understanding dialogues involving multiple participants and answering related questions. There are two main challenges for research in this field: 1) Complex discourse structure stems due to the frequent switching of speakers and changes in topics. 2)The distinct expressive intentions and speaking styles of individual speakers introduce intricate co-referential data into the dialogue. Therefore, we present the Dual-level Graph Reasoning with Key Block Decoupling (DlGR-KB) method to filter out noisy, irrelevant contexts and maintain comprehensive and accurate dialogue information flow. We first decouple the key block most relevant to the question based on a topic-aware dialogue segmentation method. To maintain the relevance of the information flow, we then employ a dual-level graph reasoning approach that integrates local and global relevance for precise and contextual awareness. At the local level, we construct a Local Interlocutor-Perceived Question Heterogeneous Graph (LIQHG) to directly connect questions with key block contents, aiding precise answer localization. Furthermore, to address the intricacies of co-reference information, we employ a global coreference-aware module (GCM) to refine the semantic logic at the representation level. Experiments conducted on the benchmark datasets demonstrate that our approach obtains consistent and significant improvements, and outperforms the performance of state-of-the-art methods.

Keywords: Multi-party Dialogue · Reading Comprehension · Graph Reasoning · Answer Extraction

1 Introduction

Traditional machine reading comprehension (MRC) typically entails extracting answers to questions from information contained within a single-scene text [22,10,25]. Multi-party dialogue reading comprehension (MDRC) differs from MRC in that it requires consideration of the entire conversation of multiple participants to understand and answer questions [27,12,30]. MDRC poses greater challenges due to the intricate relationships and knowledge backgrounds inherent

* ✉



niuq: there is any way to disable tapping from my touchpad, it's really annoying

DjViper: you could disable it alltogether

foso: that line is already there

DjViper: okay,do the next step then

niuq: yes i do,i dont even have a mouse

foso: when i control x and then say yes then enter it says error writing permission denied

DjViper: you need to do it as root, read up

foso: i do it!

Question 1: Why dose niuq want to disable tapping from touchpad ?
Answer: it's really annoying

Question 2: How does foso deal the error writing permission denied?
Answer: do it as root, read up

Fig. 1. An example of multi-party dialogue from Molweni.

in the conversation: 1)Its complex discourse structure stems from the frequent switching of speakers and changes in topics. 2)The distinct expressive intentions and speaking styles of individual speakers introduce intricate co-referential data into the dialogue. For example, as shown in Figure 1, the topics of the dialogue (marked by rectangles of various colors) change at U_2, U_3, and U_5, respectively, thereby splitting the dialogue context into four segments. The segment most relevant to *Question 1* is the first one, while the most relevant to *Question 2* is the last one. Moreover, there is a great deal of co-referential information in the context, such as "you" in U_6 and "i" in U_5 and U_7 all referring to the same speaker "foso" (marked in green).

Recently, in order to consider the dialogue characteristics, graph-based models have been proposed because graph reasoning can effectively connect the clues in discourse [24,26,31,13]. However, existing methods still face two major challenges. **Firstly**, multi-party dialogues often involve multiple participants, various topics, and frequently, numerous rounds of content. Most models still model the entire dialogue content [19,7,15], leading to the infiltration of many irrelevant topics that can cause interference. However, not all content contributes to the final response. **Secondly**, although some works consider the locally relevant content related to the question [32,16,11], the way they acquire localized content only utilizes a simple matching mechanism to extract the most critical discourse piecemeal, without considering the continuity of the dialogue and the topic information implied by the utterances; and the internal semantic relationships of each utterance within these local contents are frequently neglected, which hinders the accurate and complete perception of clues occurring in the context.

Based on the above analysis, to exclude noisy question-unrelated contexts while ensuring the comprehensiveness and accuracy of the dialogue information flow, we propose the Dual-level Graph Reasoning with Key Block Decoupling (**DlGR-KB**) method. This method not only identifies key information blocks in long dialogues while excluding redundant information but also ensures the coherence of local dialogue content and the connectivity of global semantics, thereby avoiding misunderstandings caused by incomplete information in local content. Specifically, considering the potential for misinterpretation arising from the decoupling of discrete key blocks, **we designed a Key Block Decoupling (KBD) module to ensure the coherence of the subject content**. Initially, based on the changing topics between utterances, we apply a dialogue topic segmentation model to divide long dialogue content into distinct segments. Subsequently, based on these segmented topic blocks, we train an extractor to acquire the parts most relevant to the query content. Furthermore, to ensure the relevance of the information flow within the dialogue, **we employ a dual-level graph reasoning approach to integrate both local and global informational correlations**. For the local content within the key block, we construct a heterogeneous graph, the Local Interlocutor-aware Question-Utterance Heterogeneous Graph (**LIQHG**), to establish associations between the query and the content, which can locate the answer fragments accurately. To address the complex coreference issues present in the whole dialogue content, we incorporate a Global Coreference-aware Module (abbr., **GCM**) that further refines the semantic logic within the dialogue at the representation level.

To validate the effectiveness of our method, we utilize BERT [3] and ELECTRA [2] as backbones and conduct experiments and analyses on two publicly available datasets, Molweni [12] and FriendsQA [30]. The results demonstrate three contributions of this paper:

- Our key block decoupling effectively compresses the content of conversational text, excluding irrelevant information.
- Graph reasoning with the fusion of local and global information ensures semantic relevance, leading to substantial improvements compared to graph-based baselines.
- Across different backbone models and benchmark corpora, our method demonstrates a consistent degree of improvement over other models, indicating its robustness and stability.

2 Related Work

2.1 Multi-party Dialogue Reading Comprehension

MDRC has shown more challenges than the traditional MRC. In MDRC, the shift roles of the speakers and complicated discourse relations among utterances hinder the machine from performing reading comprehension. A superficial understanding of the dialogue context is usually insufficient. Hsu et al.[9] proposed the Role-Aware Multiparty Network (RAMPNet), and experimental results demonstrated the importance of this focus. Li et al.[17] proposed the Bidirectional Information Decoupling Network (BiDeN), which utilizes temporal features in multi-round conversations for modeling, and the model is simple but effective. CADA[14] is a two-channel coding network that encodes discourse portraits and interaction relations separately, solving the problem of confusion between heterogeneous features. Li and Zhao [16] used self-supervised tasks to identify key utterance elements, their approach was typical of the early successful application of key utterances strategies to MDRC.

2.2 Graph-based Dialogue Modeling

For dialogue understanding, graphs are still a hotspot for various purposes [24,26,31]. During the development of MDRC, many researchers utilize graphs-based methods to mine deeper features in the context. Ma et al. [19] performed graph modeling for speaker-related features and discourse structure features, respectively. Gao et al. [7] applied a hierarchical graph convolutional neural network to model explicit dialogue information. Li et al.[11] first proposed a key utterances-extracting method, and constructed QuISG based on that to connect questions, speakers, and utterances. Li et al. [15] proposed a question-aware global-to-local graph reasoning approach, that implements the the progressive reasoning from the global graph to the local by a two-stage encoder network.

In this study, graphical reasoning is also the focus of our work. Different from previous work, we first perform information filtering to avoid constructing complex heterogeneous graphs over the whole context; instead, we model graphs specifically for local blocks. Additionally, when combined with global co-referential information, this dual-level graph reasoning improves the model's comprehension ability.

3 Methodology

Firstly, we show the task definition of MDRC. Given a multi-party dialogue $D = \{U_0, U_1, ..., U_N\}$ and question Q, D includes N utterances. Each U_i includes the name of the speaker S_i who issues the utterance and the concrete content C_i. Thus, the i-th utterance can be denoted as $U_i = \{S_i, C_i\}$, and C_i is an l_i-length sequence consisting of words w_{ij}, i.e., $C_i = \{w_{i1}, w_{i2}, ..., w_{ij}, ..., w_{il_i}\}$. MDRC aims at extracting the answer span $Answer_{span} = [Start, End]$ from D.

The general architecture of our method is shown in Figure 2. The model predicts answers through dual-level graph reasoning that includes local and global information incorporating. Local parsing can effectively exclude irrelevant dialogue topics based on key block content, while global parsing can effectively understand complex reference information in dialogue based on global reference association. Next, we first show the task definition and then introduce the details of our method.

3.1 Encoding Layer

For a better contextual representation, we first use a Pre-trained Language Model (PrLM) to encode the question and context of the dialogue. For the dialogue context $D = \{U_0, U_1, ..., U_N\}$ and question Q, we concatenate them in the form of $[CLS]\, Q\, [SEP]\, D\, [SEP]$. However, as we need to deal with independent utterance units, we insert $[SEP]$ between each pair of neighboring utterances to facilitate the division. Finally, we process the concatenated sequence in the form of $[CLS]\, Q\, [SEP]\, U_0\, [SEP]\, U_1\, [SEP]\, ...U_N\, [SEP]$. We input the concatenated sequence into the PrLM. The output of the PrLM is an initial contextual representation of each token, denoted as $H \in R^{n \times d}$, where n represents the length of the input sequence and d represents the dimension of the hidden states.

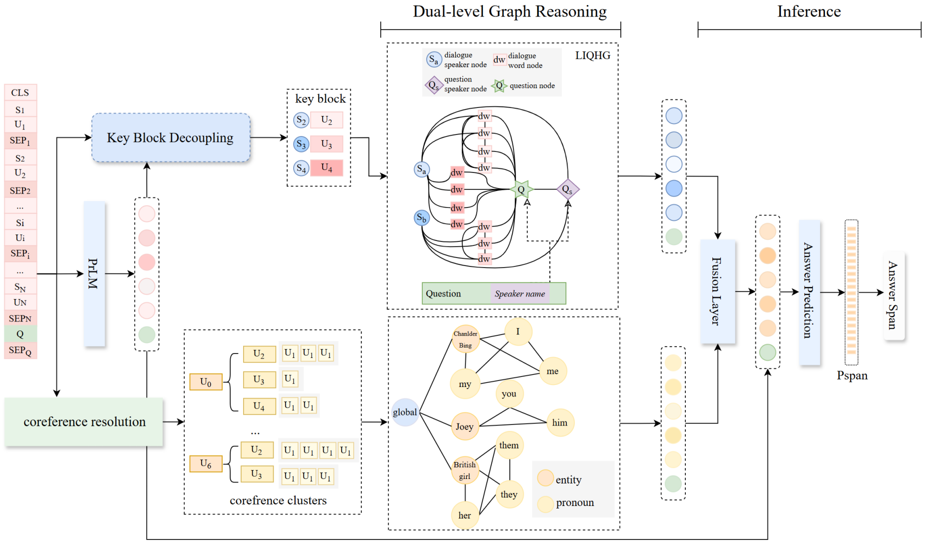


Fig. 2. An overview of DlGR-KB.

3.2 Key Block Decoupling

There is a constant change of subject matter in multi-party dialogue contexts, and neighboring utterances may contain different dialogue topics. We design a key block decoupling module to prioritize essential dialogue content, enabling the model to consciously attend to crucial dialogue segments.

In order to ensure that the extracted key content remains continuous and to prevent any comprehension deviations caused by the discontinuity of semantic content, we employ a dialogue topic segmentation method [4] to decouple the dialogue. Specifically, SimCSE [5] initializes the topic encoder first to obtain the topic representation for each discourse. Subsequently, it selects the Next Sentence Prediction (NSP) BERT [3] as the coherence encoder to calculate the coherence scores for each utterance pair. After computing the relevance scores based on the topic representation and coherence scores, TextTiling [8] is applied to determine the segment boundaries. Additionally, two self-supervised tasks are utilized: the Neighboring Utterance Matching (NUM) task for training the topic encoder and the Relevance Modeling (RM) task for training both encoders.

Figure 3 shows the framework of the KBD. We first use the trained model on the DialSeg711 dataset[29] for dialogue segmentation, facilitating the extraction and disentanglement of key blocks. For a dialogue $D = \{U_0, U_1, ..., U_N\}$, we obtain segment boundaries $B = \{b_0, b_1, ..., b_n\}$, where utterances between b_0 and b_1 ((inclusive of b_0) from a block, and n represents the total number of blocks. Subsequently, we train the key block extractor using our training datasets. Leveraging the contextualized representations H from Section 3.2, we collect the $[SEP]$ token representations and employ them as utterance representations within the dialogue context. We initialize N utterance representations $H_U = \left\{H_{U_i} \in R^d\right\}_{i=1}^{N}$ and the question representation $H_Q \in R^d$. Each H_{U_i} is paired

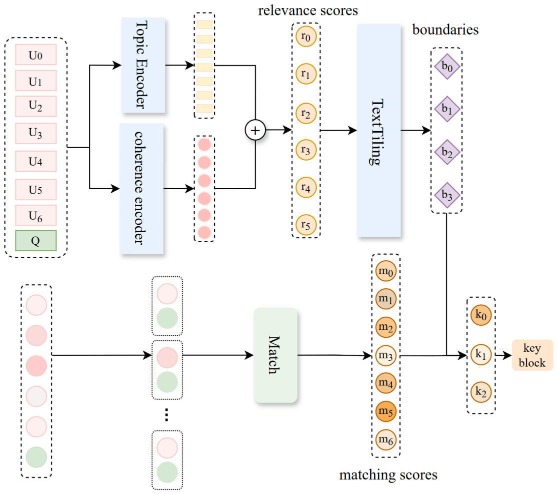


Fig. 3. The framework of Key Block Decoupling module.

with H_Q, and we define a matching function $Match\,(X, Y, \sigma)$ inspired by the heuristic matching mechanism [21] to calculate the matching scores M_{U_i} for each utterance-question pair. Based on $M = \{M_{U_0}, M_{U_1}, ..., M_{U_N}\}$ and B, we initialize a PrLM and use it to compute the average matching score for each block and derive the probability distribution of key blocks as $P_S^{pred} = \{P_0, P_1, ..., P_n\}$. For each block, if any utterances within it contain the answer, we designate its index $S^{target} = i$ as the target. Consequently, the training objective for the key block extractor in dialogue is:

$$L_s = -\log\left(P_S^{pred}\left[S^{target}\right]\right). \tag{1}$$

Using the trained extractor, we extract the key block $K = \{U_0, U_1, ..., U_k\}$ containing k utterances from the dialogue context.

3.3 Dual-level Graph Reasoning

Due to the complexity of information in a multi-party dialogue, which includes multiple topics and interlocutors, and also maintains the coherence of semantic information flow, it is necessary to maintain a global association of information while decoupling the content to exclude irrelevant topics. DlGR-KB establishes a dual-level graph reasoning pathway. Based on local key blocks, we construct the Local Interlocutor-aware Question-Utterance Heterogeneous Graph (LIQHG). For globally complex reference relationships, we integrate the Global Coreference-aware Module (GCM).

LIQHG For the key block $K = \{U_0, U_1, ..., U_k\}$ with k utterance units extracted, each U_i includes the name of speaker S_i who issues the utterance and the concrete content C_i. Thus, the i-th utterance can be denoted as $U_i = \{S_i, C_i\}$. To narrow down the model's understanding of the question scope, the interlocutor plays a crucial role in defining the boundaries of the issue to some extent. Therefore, when constructing the graph, we associate the interlocutors within the key block with those mentioned in the question by linking their content, thereby building a heterogeneous graph. We utilize the speakers in question S_q, the question Q and the speakers and concrete content of K to construct the Interlocutor-aware Question-Utterance Heterogeneous Graph $G_k = (V^k, M^k)$, where V^k is the set of nodes, and M^k denotes the adjacent matrix of edges.

-Graph Initialization To ensure the relationships between information, we have constructed four different types of nodes. The question node is represented by the question word in the question (e.g., "what", "why"), and we initialize it by applying mean pooling to the representation of the question words: $v.q = mean\left(H_Q\left[\textbf{question word}\right]\right)$. Considering the speakers mentioned in the question can help the model identify which speakers and their interactions are most crucial for answering the question. We use neuralcoref [32] to recognize speaker names appearing in the question and select those that also appear in the dialogue context. Then, we utilize these speaker names to initialize the question speaker nodes: $v.qs = H_Q\left[\textbf{speaker names}\right]$. The speakers of utterances are an important part of the dialogue context features. We extract the speakers of all utterances from the key segment and initialize the dialogue speaker nodes as $v.ds = H_U\left[\textbf{speaker names}\right]$. As the most likely source of the answer, all words in the key segment are considered as dialogue word nodes for constructing the graph, initialized as $v.dw = H_U\left[C_0, C_1, ..., C_k\right]$, where $\left[C_0, C_1, ..., C_k\right]$] represents the concrete content of the key block.

We use the adjacency matrix M^k to define the edges between nodes, and M^k is symmetric. The connection rules between nodes are established as follows: all word nodes within the same key block are connected, as well as to the speaker nodes of their respective utterances and the question node; question speaker nodes are connected, to all dialogue speaker nodes of the key block, and the question node; all nodes also include self-connecting edges. In the matrix M^k, nodes for which an edge exists have a value of 1, and 0 otherwise. The framework for constructing the graph is shown in Figure 2.

-Graph Reasoning After constructing LIQHG, we utilize the Node-Type Realized Graph Attention Network to enhance the knowledge transfer of nodes in it. The representation of each node $v_i^k \in V^k$, denoted by h_i, can be obtained from the encoding H. For node v_i^k and v_j^k, we calculate the node type realized attentive weight and aggregate the weighted message for v_i^k by the formula:

$$\begin{aligned} c_{ij} &= a\left[\left[h_{v_i^k}^{t-1} || r_{v_i^k.t}\right] w_q \left[h_{v_j^k}^{t-1} || r_{v_j^k.t}\right] w_k\right]^T, \\ \alpha_{ij} &= \frac{\exp\left(LRELU\left(c_{ij}\right)\right)}{\sum_{v_o^k \epsilon N_{v_i^k}} \exp\left(LRELU\left(c_{io}\right)\right)}, \\ h_{v_i^k}^{l.head} &= ELU\left(\sum_{v_o^k \epsilon N_{v_i^k}} \alpha_{ij} h_{v_o^k}^{l-1} W_o\right), \end{aligned} \tag{2}$$

where $r_{v_i^k.t} \in R^{1\times4}$ is a one-hot vector denoting the node type of v_i^k, $N_{v_i^k}$ represents the neighbors of node v_i^k, and $a \in R^{1\times 2d_{head}}$, $W_q \in R^{(d_{head}+4)\times d_{head}}$,

$W_k \in R^{(d_{head}+4) \times d_{head}}$, $W_o \in R^{d_{head} \times d_{head}}$ are trainable parameters. $h^{l.head}_{v_i^k}$ is the representation of the node v_i^k processed by the single-head attention in the l-th graph attention layer, then we concatenate weighted messages from all heads and obtain the node representation ${h_i}^l$ in the l-th graph attention layer. After L multi-head GAT layers, collecting these for all nodes gives us the role behavior-aware feature representation $H_k = \left\{h_i^L \in R^d\right\}_{i=1}^{m}$, where m is the total number of nodes.

GCM Since each speaker possesses a unique style of speaking, intention, and habit of expression, this results in a significant amount of intricate co-reference information within the dialogue. A pronoun such as "it" may necessitate multi-skip reasoning across the utterances of different interlocutors, thereby increasing the difficulty of comprehension. By connecting the referring objects in the dialogue through Coreference Resolution, the model can capture the specific semantic details of the dialogue at a fine-grained level. Based on this, we construct a global coreference-aware graph $G_c = (V^c, E^c, M^c)$, where V^c represents the set of nodes generated from the co-reference information, E^c represents the set of edges, and M^c is the corresponding adjacency matrix.

-Extracting Coreference Clusters We use the coreference resolution toolkit (such as FastCoref [23]) to identify referential entities and their corresponding expressions in the dialogue and extract co-reference clusters. For instance, the text “Chandler Bing: And I love the milk! But, I’m not gonna some British girl to move in with me! Joey, do you welcome them ?” will generate three coreference clusters: *[Chanlder Bing: Chanlder Bing, I, I, me]*, *[British girl: British girl, them]*, and *[Joey: Joey, you]*. In the first cluster, the speaker *Chanlder Bing* is recognized as the root entity, and *(Chanlder Bing, I, I, me)* is recognized as pronouns related to *Chanlder Bing*. For the entire dialogue D, it will be parsed to generate multiple coreference clusters, where each cluster contains a root entity and several pronouns that share the same meaning as the root entity.

-Graph Initialization and Reasoning Obviously, since coreference perception is token-level information perception, we treat each token in the input as a node. Furthermore, we partition nodes from the same co-referring cluster into a subgraph and construct edges e^c_{ij} for each pair of nodes $v^c_i \rightarrow v^c_j$. Specifically, we set the value of e^c_{ij} to 1 to indicate the existence of these edges in the subgraph, and to $-\infty$ otherwise. This forms the subnode set V^c_i and the subedge set E^c_i. which strengthens the connection between coreference terms. A dialogue containing $\mathcal{C}$ coreferenced clusters will result in $\mathcal{C}$ such subgraphs. Additionally, we consider the $[CLS]$ token as the global root node, connecting it to the root node of each subgraph, thereby establishing a global connection for the coreference information contained in the entire dialogue.

The representation of each node h_i in V^c, can be obtained from the encoding H by the subscripts t_i (where $i = 1, 2, \ldots, m$) recorded during the preprocessing stage. Graph Attention Network (GAT) [28] is often chosen for building graph neural models due to its adaptive neighbor aggregation, excellent parallel computing ability, and strong generalization ability for handling homogeneous and heterogeneous graphs. Thus we apply GAT to propagate and aggregate messages between nodes in the graph. The calculation is as follows:

$$
\begin{aligned}
h_i^{(l+1)} &= \Sigma_{j\in N_i}\alpha_{ij}^{(l)}W^{(l)}h_j^{(l)}, \\
\alpha_{ij}^{(l)} &= \frac{\exp\left(f\left(\left[h_i^{(l)};h_j^{(l)}\right],W_0^{(l)}\right)\right)}{\Sigma_{h_k\in N_i}\exp\left(f\left(\left[h_i^{(l)};h_k^{(l)}\right],W_0^{(l)}\right)\right)},
\end{aligned}
\tag{3}
$$

where N_i represents the neighbors of node v_i^c. α_{ij} measures the attention between h_i and h_j. GAT computes this attention at each layer l by concatenating h_i and h_j, passing them through a LeakyReLU [20] activation function, and multiplying by trainable parameters $W^{(l)}$ and $W_0^{(l)}$. The whole process can be summarized as equation 4. This process leads to a highly adaptive representation of each node, contributing to the effective modeling of the discourse dependency structure.

$$
h^{(l+1)} = CorefGAT\left(V^c, E^c, M^c\right). \tag{4}
$$

$h^{(l+1)}$ represents the hidden state of all nodes in $(l+1)$-th layer. After L Coref-GAT layers, we obtain the dependency-aware co-reference feature representation $H_c = \left\{h_i^L \in R^d\right\}_{i=1}^m$,where m is the total number of nodes.

3.4 Answer Prediction and Training

After the dual-level graph reasoning process, we obtain two feature representations: one focusing on localized crucial utterances $(H_k)$and the other incorporating global coreference information (H_c). To effectively fuse these multi-features, we employ the attention mechanism for bidirectional integration, combining internal semantic details with integral co-reference information to form a richer feature representation (H'). The probability distributions of the answer span are then calculated using the following formula.

$$
\begin{aligned}
P_{start} &= soft\max\left(W_{start}^T H'\right), \\
P_{end} &= soft\max\left(W_{end}^T H'\right),
\end{aligned}
\tag{5}
$$

where W_{start}^T and W_{end}^T are trainable weight vectors. Given the true labels of the answer span $[A_s, A_e]$, we adopt a cross-entropy function to train our model:

$$
L_{span} = -\left(\log\left(P_{start}\left[A_s\right]\right) + \log\left(P_{end}\left[A_e\right]\right)\right). \tag{6}
$$

If the dataset contains unanswerable questions, we predict whether a question is answerable using the formula $p_a = sigmoid(W^T H\left[CLS\right] + b)$, where W^T and b are trainable weights vectors of size R^d. Given the ground truth of answerability $q_a \in \{0,1\}$, we apply cross-entropy to calculate the answerable loss as follows:

$$
L_a = -\left((1-q_a) * \log\left(1-p_a\right) + q_a * \log\left(p_a\right)\right). \tag{7}
$$

In this way, the overall training objective of our method is $L = L_{span} + L_a$.

Table 1. Overview of Molweni and FriendsQA.

Datasets	Molweni			FriendsQA		
	Train	Dev	Test	Train	Dev	Test
Dialogues	8,771	883	100	977	122	123
Utterances	77,374	7,823	845	21,607	2,847	2,336
Questions	24,682	2,513	2,871	8,535	1,010	1,065

4 Experiments

In this section, we evaluate the performance of our method on the two benchmark corpora, and explore the following questions:

- **RQ1**: Does the DlGR-KB model produce consistent results compared to other baseline methods?
- **RQ2**: Does the dual-level graph reasoning in DlGR-KB contribute effectively to its performance?
- **RQ3**: Can the key block localization effectively identify continuous segments related to the answer, serving the function of compressing the dialogue content length?
- **RQ4**: How does the performance of DlGR-KB change in response to different types of questions?

4.1 Experimental Setting

Datasets. Following the previous work [12,7,15], we experiment on two benchmark corpora, Molweni and FriendsQA. Molweni is a large-scale dataset derived from Ubuntu Chat Corpus. FriendsQA, derived from the popular TV show "Friends", contains more utterances per dialogue. The details of the datasets are shown in Table 1.

Backbone and Hyperparameter Settings. To verify the stability of our model, we construct it using different PrLMs as backbones, including BERT-Base-Uncased [3] and ELECTRA [2]. Additionally, we set different hyperparameters for each model and use the AdamW optimizer [18] to perform fine-tuning. For Molweni dataset experiments, batch sizes are 2 for BERT-Base and 4 for ELECTRA, with learning rates 2.5e-5 and 1.2e-5. For FriendsQA, batch sizes are 2 for BERT and 4 for ELECTRA, with learning rates 1.8e-5 and 4e-6. Exact Matching (EM) and F1 scores are the metrics we use to measure performance.

Compared Models[3]. To verify the effectiveness of our model, we compare different types of baselines, including direct modeling methods BiDeN[17],CADA[14], graph-based methods DADgraph[13], Dis-QueGC[7], ESA[19], GLGR[15], M-HGN[6] and local analysis methods that extract relevant content in combination with question SKIDB[16], QUISG[11], DGKC[1].

[3] The performance results for comparing models are all sourced from publicly available papers.

Table 2. Experimental results on Molweni and FriendsQA.

	Molweni				FriendsQA			
	EM	F1	EM	F1	EM	F1	EM	F1
Model	*BERT*		*ELECTRA*		*BERT*		*ELECTRA*	
Baseline	45.7	60.3	56.8	70.6	42.7	60.0	57.0	74.8
DADgraph[13]	46.5	61.5	-	-	-	-	-	-
SKIDB[16]	49.2	64.0	58.0	72.9	46.9	63.9	55.8	72.3
BiDeN[17]	48.1	63.2	-	-	-	-	-	-
Dis-QueGCN[7]	47.7	63.6	-	-	-	-	-	-
ESA[19]	49.7	64.4	58.6	72.2	47.0	63.0	58.7	75.4
GLGR[15]	48.2	64.4	59.2	73.6	47.0	64.3	59.8	**77.2**
CADA[14]	-	-	59.8	73.6	-	-	59.2	76.7
QUISG[11]	-	-	59.3	72.9	-	-	57.8	75.2
DGKC[1]	49.9	63.9	-	-	-	-	-	-
M-HGN[6]	49.2	63.9	-	-	-	-	-	-
DlGR-KB	**50.2**	**65.1**	**60.3**	**73.9**	**47.6**	**64.4**	**60.9**	**77.2**

The bold values indicate the best result.

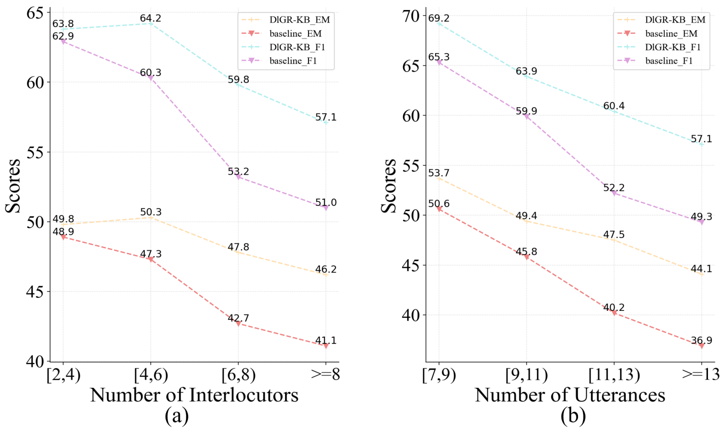


Fig. 4. Performance of DlGR-KB and baseline(*BERT*) on Molweni with different interlocutor and utterance numbers.

4.2 Main Results (RQ1)

Comparison with Baselines. Table 2 presents the results of the comparison experiments between our model and related models. Overall, graph-based modeling approaches have yielded superior results. The method proposed in this paper demonstrates significant improvement compared to other methods. Among them, SKIDB and QUISG also achieved good performance in Molweni, indicating that extracting content related to the question is beneficial for answer prediction. However, their less pronounced performance also underscores the importance of semantic coherence in dialogue, especially in the FriendsQA dataset where content and speaker language style characteristics vary more prominently. Therefore, the modeling approach of GLGR, DGKC, and M-HGN , which integrates linguistic relationships, has gained a notable advantage. Our method, DlGR-KB,

Table 3. Results of ablation study.

Model	Molweni		FriendsQA	
	EM	F1	EM	F1
DlGR-KB(BERT)	50.2	65.1	47.6	64.4
w/o KBD	49.3	64.4	46.8	63.5
w/o GCM	48.6	63.2	46.1	62.7
baseline	45.7	60.3	42.7	60.0
+KBD	47.8	60.5	44.9	61.8
+LIQHG	49.0	63.4	46.2	62.7
+GCM	49.2	63.6	46.9	63.5

fully considers both local information and maintains a certain degree of content relevance, as well as the overall semantic coherence of the dialogue. Consequently, it achieved the best results across both backbones and datasets

The Impact of Interlocutor Number. To further validate the effectiveness of DlGR-KB in multi-party dialogues, we divided the Molweni dataset into four parts based on the number of interlocutors involved in the dialogues: 1) 2-4 interlocutors, 2) 4-6 interlocutors, 3) 6-8 interlocutors, and 4) no less than 8 interlocutors. The results are illustrated in Figure 4 (a). For smaller numbers of interlocutors, the baseline and DlGR-KB (with BERT as the backbone) exhibit similar performance. However, when the dialogue includes a large number of interlocutors, performance degrades for both the baseline and our model, as both are affected by a significant amount of noise. Notably, as the number of interlocutors increases, the performance gap between the baseline and our model widens.

4.3 Ablation Study (RQ2)

We perform an ablation analysis to evaluate the impact of each module on the overall performance of our model. As the LIQHG module is built upon the Key Block Decoupling (KBD) module, only the results of removing the KBD and GCM are shown in Table 3. By removing these modules, we observe a decrease in model performance. Additionally, we individually add those three modules to the baseline ($BERT$), which more intuitively shows their contributions to the model. As shown in Table 3, all three modules improve the final performance of our model, with the GCM module contributing the most, resulting in nearly a 3.5% increase in EM-score and a 3.3% increase in F1-score on the Molweni dataset. This is because there is a significant amount of co-reference information in the dialogue contexts, and parsing this information allows the model to better grasp the details of the dialogue and enhance comprehension. When the LIQHG module is introduced based on the KBD, it results in an additional 1.2% increase in the EM-score and a 2.9% increase in the F1-score for the model. Similar trends are also observed in the FriendsQA dataset.

4.4 Key Block Analysis (RQ3)

The Accuracy of Key Block Decoupling. To test the accuracy of key block decoupling and its ability to encompass the location of the answer, we conduct a separate analysis of the results for this component. During the training of the

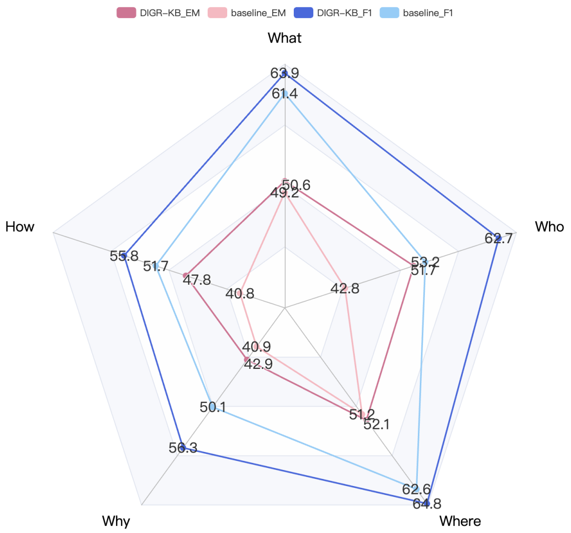


Fig. 5. Results on questions with different interrogative words.

KBD, we organize the dataset by segmenting it based on dialogue topics and identifying the modules that contain the answers as the key block. In the Molweni dataset, after training, the accuracy of the KBD component reaches 86.4%, and in the FriendsQA dataset, the accuracy reaches 90.2%.

The Impact of Utterance Number. Figure 4 (b) illustrates the impact of different numbers of utterances on DlGR-KB. By segmenting the data based on the number of utterances, we observe that when the number of turns is relatively low, the gap between our method and the baseline is narrow. This suggests that when the dialogue content is inherently brief, the advantages of KBD and dual-level graph reasoning are not pronounced. As the number of turns increases, our method begins to demonstrate its superiority. It not only filters out some irrelevant information and interference but also leverages the semantic coherence of long texts, which plays a crucial role.

4.5 Analysis on Different Interrogative Words (RQ4)

To delve deeper into our model's performance, we evaluate its improvement on questions featuring various interrogative words, which direct the model's focus within the discourse. Specifically, we analyze five primary question types: *Who*, *How*, *Why*, *Where*, and *What*, categorizing "Whose" under "Who". Figure 5 displays the scores of BERT and our model for each type. Key insights include: (i) Our model enhances performance across all question types, aligning with the main experiment findings. (ii) *Who* questions show the greatest improvement (8.9 EM, 9.5 F1), attributed to KBD and Dual-level Graph Reasoning, which focus on relevant speaker and utterance content, enabling word-level modeling and deep understanding of speaker behavior. Additionally, modeling co-reference information benefits most questions, enhancing global reasoning. (iii) *How* and *Why* questions pose the greatest challenge due to their demand for deep logical reasoning.

4.6 Case Study

In this section, we choose a case based on the test set of Molweni for analysis. Figure 6 shows one of the results. When the prediction result of the baseline does not match the ground truth, our model can predict the accurate answer span. We first segment the dialogue into four blocks. The four utterances U_0-U_4 in the first block share a common topic “program”, and the word “supports cam” appears frequently in the question and utterances, resulting in the highest average match score for the first block. Moreover, there is a great deal of co-referential information throughout the dialog, such as “ActionParsnip” and “i” in U_3, and “mataks” and “im” in U_5 (highlighted in green and pink in the context). Our model utilizes all of the above information to get the correct answer in the end.

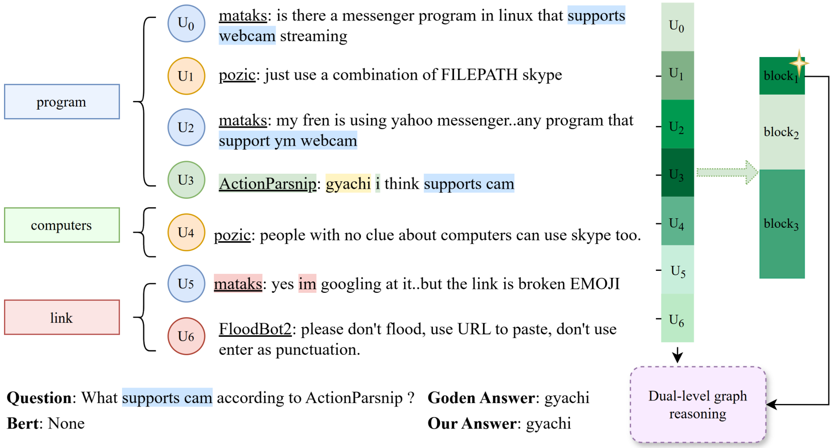

program
computers
link
U0
U1
U2
U3
U4
U5
U6
mataks: is there a messenger program in linux that supports webcam streaming
pozic: just use a combination of FILEPATH skype
mataks: my fren is using yahoo messenger..any program that support ym webcam
ActionParsnip: gyachi i think supports cam
pozic: people with no clue about computers can use skype too.
mataks: yes im googling at it..but the link is broken EMOJI
FloodBot2: please don't flood, use URL to paste, don't use enter as punctuation.
block1
block2
block3
Dual-level graph reasoning
Question: What supports cam according to ActionParsnip ?
Bert: None
Goden Answer: gyachi
Our Answer: gyachi

Fig. 6. A case study for DIGR-KB. The heatmap shows the probability distribution of the key blocks.

5 Conclusion

In this work, we propose a dual-level graph reasoning with key block decoupling method towards MDRC. It explores critical cues for extracting answer information from both local and global perspectives. Given the challenges in locating answers within long dialogue texts, our method organizes and segments the dialogue content, decoupling the key block closely related to the question while ensuring the coherence of the decoupled content. We construct a heterogeneous graph for reasoning based on the information within the key block, establishing connections with the question’s scope. However, due to the inherent semantic relatedness of the overall dialogue content, relying solely on local information can lead to incomplete understanding or information loss. To overcome this issue, we integrate global referential information to restore any missing content in the local context. Ultimately, the experimental results and analysis fully demonstrate the effectiveness of our proposed method.

6 Acknowledgements

Xiabing Zhou is the corresponding author. This research is supported by the National Natural Science Foundation of China (62176174) and the Priority Academic Program Development of Jiangsu Higher Education Institutions.

References

1. Cao, R., Zhou, X., Zhou, G.: Structure and behavior dual-graph reasoning with integrated key-clue parsing for multi-party dialogue reading comprehension. In: NLPCC 2024, Hangzhou, China. pp. 162–174. Springer (2024)
2. Clark, K., Luong, M.T., Le, Q.V., Manning, C.D.: Electra: Pre-training text encoders as discriminators rather than generators. In: ICLR 2020, Addis Ababa, Ethiopia. OpenReview.net (2020)
3. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: NAACL-HLT 2019, Minneapolis, MN, USA. pp. 4171–4186. Association for Computational Linguistics (2019)
4. Gao, H., Wang, R., Lin, T., Wu, Y., Yang, M., Huang, F., Li, Y.: Unsupervised dialogue topic segmentation with topic-aware utterance representation. CoRR **abs/2305.02747** (2023)
5. Gao, T., Yao, X., Chen, D.: Simcse: Simple contrastive learning of sentence embeddings. In: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic. pp. 6894–6910. Association for Computational Linguistics (2021)
6. Gao, X., Zhou, X., Cao, R., Zhang, M.: TGAT-DGL: triple graph attention networks on dual-granularity level for multi-party dialogue reading comprehension. In: IJCNN 2024, Yokohama, Japan. pp. 1–8. IEEE (2024)
7. Gao, X., Zhou, X., Zhang, M.: A multi-information perception based method for question answering in multi-party conversation. Acta Scientiarum Naturalium Universitatis Pekinensis **59**(01), 21–29 (2023)
8. Hearst, M.A.: Texttiling: Segmenting text into multi-paragraph subtopic passages. Comput. Linguistics **23**(1), 33–64 (1997)
9. Hsu, J.H., Shen, P.W., Su, H.T., Chang, C.H., Yeh, J.F., Hsu, W.H.: Role aware multi-party dialogue question answering. In: ICASSP 2021, Toronto, ON, Canada. pp. 7813–7817. IEEE (2021)
10. Lai, G., Xie, Q., Liu, H., Yang, Y., Hovy, E.H.: RACE: large-scale reading comprehension dataset from examinations. In: Palmer, M., Hwa, R., Riedel, S. (eds.) EMNLP 2017, Copenhagen, Denmark. pp. 785–794. Association for Computational Linguistics (2017)
11. Li, J., Yu, M., Meng, F., Lin, Z., Fu, P., Wang, W., Zhou, J.: Question-interlocutor scope realized graph modeling over key utterances for dialogue reading comprehension. In: ACL 2023, Toronto, Canada. pp. 4956–4968. Association for Computational Linguistics (2023)
12. Li, J., Liu, M., Kan, M.Y., Zheng, Z., Wang, Z., Lei, W., Liu, T., Qin, B.: Molweni: A challenge multiparty dialogues-based machine reading comprehension dataset with discourse structure. In: COLING 2020, Barcelona, Spain (Online). pp. 2642–2652. International Committee on Computational Linguistics (2020)
13. Li, J., Liu, M., Zheng, Z., Zhang, H., Qin, B., Kan, M.Y., Liu, T.: Dadgraph: A discourse-aware dialogue graph neural network for multiparty dialogue machine reading comprehension. In: IJCNN 2021, Shenzhen, China. pp. 1–8. IEEE (2021)
14. Li, Y., Zou, B., Fan, Y., Dong, M., Hong, Y.: Coreference-aware double-channel attention network for multi-party dialogue reading comprehension. In: IJCNN 2023, Gold Coast, Australia. pp. 1–8. IEEE (2023)

15. Li, Y., Zou, B., Fan, Y., Li, X., Aw, A., Hong, Y.: Glgr: Question-aware global-to-local graph reasoning for multi-party dialogue reading comprehension. In: EMNLP 2023, Singapore. pp. 1817–1826. Association for Computational Linguistics (2023)
16. Li, Y., Zhao, H.: Self-and pseudo-self-supervised prediction of speaker and key-utterance for multi-party dialogue reading comprehension. In: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic. pp. 2053–2063. Association for Computational Linguistics (2021)
17. Li, Y., Zhao, H., Zhang, Z.: Back to the future: Bidirectional information decoupling network for multi-turn dialogue modeling. In: EMNLP 2022, Abu Dhabi, United Arab Emirates. pp. 2761–2774. Association for Computational Linguistics (2022)
18. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: ICLR 2019, New Orleans, LA, USA. OpenReview.net (2019)
19. Ma, X., Zhang, Z., Zhao, H.: Enhanced speaker-aware multi-party multi-turn dialogue comprehension. IEEE/ACM Transactions on Audio, Speech, and Language Processing **31**, 2410–2423 (2023)
20. Maas, A.L., Hannun, A.Y., Ng, A.Y.: Rectifier nonlinearities improve neural network acoustic models **30**(1), 3 (2013)
21. Mou, L., Men, R., Li, G., Xu, Y., Zhang, L., Yan, R., Jin, Z.: Natural language inference by tree-based convolution and heuristic matching. In: ACL 2016, Berlin, Germany. The Association for Computer Linguistics (2016)
22. Nallapati, R., Zhou, B., dos Santos, C.N., Gülçehre, Ç., Xiang, B.: Abstractive text summarization using sequence-to-sequence rnns and beyond. In: CoNLL 2016, Berlin, Germany. pp. 280–290. ACL (2016)
23. Otmazgin, S., Cattan, A., Goldberg, Y.: F-coref: Fast, accurate and easy to use coreference resolution. In: AACL/IJCNLP 2022 - System Demonstrations, Taipei, Taiwan. pp. 48–56. Association for Computational Linguistics (2022)
24. Qin, L., Li, Z., Che, W., Ni, M., Liu, T.: Co-gat: A co-interactive graph attention network for joint dialog act recognition and sentiment classification. In: AAAI 2021. pp. 13709–13717. AAAI Press (2021)
25. Rajpurkar, P., Zhang, J., Lopyrev, K., Liang, P.: Squad: 100, 000+ questions for machine comprehension of text. In: EMNLP 2016, Austin, Texas, USA. pp. 2383–2392. The Association for Computational Linguistics (2016)
26. Shen, W., Wu, S., Yang, Y., Quan, X.: Directed acyclic graph network for conversational emotion recognition. In: ACL/IJCNLP 2021. pp. 1551–1560. Association for Computational Linguistics (2021)
27. Sun, K., Yu, D., Chen, J., Yu, D., Choi, Y., Cardie, C.: DREAM: A challenge dataset and models for dialogue-based reading comprehension. CoRR **abs/1902.00164** (2019)
28. Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y., et al.: Graph attention networks. stat **1050**(20), 10–48550 (2017)
29. Xu, Y., Zhao, H., Zhang, Z.: Topic-aware multi-turn dialogue modeling. In: AAAI 2021, Virtual Event. pp. 14176–14184. AAAI Press (2021)
30. Yang, Z., Choi, J.D.: Friendsqa: Open-domain question answering on TV show transcripts. In: SIGdial 2019, Stockholm, Sweden. pp. 188–197. Association for Computational Linguistics (2019)
31. Yasunaga, M., Ren, H., Bosselut, A., Liang, P., Leskovec, J.: QA-GNN: Reasoning with language models and knowledge graphs for question answering. In: NAACL-HLT 2021, Online. pp. 535–546. Association for Computational Linguistics (2021)
32. Zhang, Z., Li, J., Zhao, H.: Multi-turn dialogue reading comprehension with pivot turns and knowledge. IEEE ACM Trans. Audio Speech Lang. Process. **29**, 1161–1173 (2021)