Information Fusion 110 (2024) 102469

Contents lists available at ScienceDirect

Information Fusion

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

Full length article

ChatMDG: A discourse parsing graph fusion based approach for multi-party dialogue generation

Jingyang Li, Shengli Song *, Yixin Li, Hanxiao Zhang, Guangneng Hu

School of Computer Science and Technology, Xidian University, Xi'an, 710126, Shaanxi, China

ARTICLE INFO

Keywords:
Multi-party dialogue
Dialogue generation
Discourse parsing
Semantic-enriched graph
Large language models

ABSTRACT

Comprehending multi-party dialogue generation poses a challenge due to intricate speaker interactions, where multiple participants engage in a dynamic exchange of questions and responses, assuming diverse roles such as speaker, receiver, and observer, with these roles evolving across conversational turns. Most existing research on multi-party dialogue generation only considers semantic information contained in each sentence and does not take into account the dialogue flow information implicit in multi-role interaction, leading to difficulties in accurately understanding the dialogue state in multi-party dialogue. To fill these gaps, we introduce an information fusion based approach for **M**ulti-party **D**ialogue **G**eneration named **ChatMDG**, which integrates role interaction into a semantic-enriched graph with context-based embeddings to cooperatively capture both global and local information in multi-party dialogue. Specifically, we proposes a graph-based network to represent the complex role-interaction dialogue structure for discourse parsing and then designs the dialogue flow encoding method to fuse role-interaction information with semantic states effectively. Furthermore, ChatMDG presents interaction strategies to correspondingly generate reactive and proactive utterances based on the fused embeddings, which lead to more dialogue coherence and user engagement. Experimental results show that ChatMDG significantly improves the accuracy of the multi-party response generation task, especially in complex scenarios with multiple interactions.

1. Introduction

Due to the widespread adoption of ChatGPT and other Large Language Models(LLMs) [1], dialogue generation, a challenging task in artificial intelligence, has emerged as a prominent area of research interest, leading to numerous valuable contributions and breakthroughs. The goal is to generate natural language responses that can naturally integrate into the dialogue interaction process [2]. Multi-party dialogue, which involves two or more participants, allocates each utterance to a specific role. The prevalence of multi-party dialogues in various practical scenarios, such as social group chats, community forums, and meetings, highlights the significance of researching technology for multi-party dialogue generation. This endeavor empowers dialogue systems to comprehend the intricate interplay between diverse roles and the unique developmental context inherent in multi-party dialogues. Ultimately, advancing multi-party dialogue generation technology enables dialogue systems to respond effectively and judiciously in multi-user interaction scenarios.

Navigating multi-party dialogues involves a complex interaction nature, as participants can respond to any role, resulting in a concurrent dialogue flow represented as a graph-based network structure [3]. At each time step, the nodes (roles) and edges (relations) in this graph have the potential to influence the direction of future dialogue flow. Therefore, how to fuse the role states is essential for the dialogue system to track the multi-party dialogue states, comprehend the dialogue context, navigate system interaction, and generate a valuable, satisfying response. However, existing multi-party dialogue models continue to rely on sequential embeddings and frequently overlook role-specific interactions [4]. Some approaches in the realm of multi-party dialogue modeling adopt deep sequential or tree structures to represent the dialogue context [5,6]. While these tree-based approaches provide a structural representation of the conversation flow, they often struggle to accurately capture the intricate and dynamic nature of the dialogue state, impeding coherence and engagement in the multi-party generation task.

Fig. 1 illustrates a multi-party dialogue generation scenario involving multiple users and the system. The key feature of this task is that the system assumes the perspective of one of the roles in the dialogue and interacts with the other roles. Firstly, the system must regulate its

* Corresponding author.
E-mail addresses: jylee@stu.xidian.edu.cn (J. Li), shlsong@xidian.edu.cn (S. Song), liyixin@stu.xidian.edu.cn (Y. Li), zhanghx@stu.xidian.edu.cn (H. Zhang), huguangneng@xidian.edu.cn (G. Hu).

https://doi.org/10.1016/j.inffus.2024.102469
Received 31 January 2024; Received in revised form 9 April 2024; Accepted 10 May 2024
Available online 13 May 2024

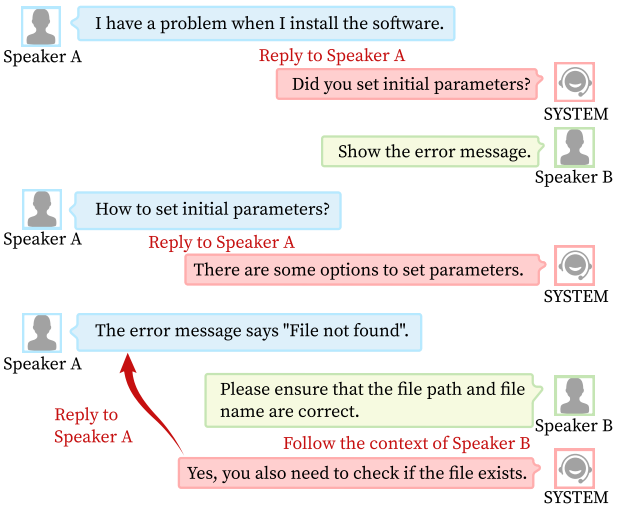

Fig. 1. Example Of Multi-party Dialogue Generation Task. The pink rectangles represent utterances generated by the System, while Speakers A and B represent users. Arrows represent the semantic relations of the System's reply utterances, forming a graph structure for multi-party dialogue.

interaction mode due to the non-round-robin nature of interactions in multi-party dialogues. Roles selectively respond to utterances based on their relevance or interest. Secondly, To determine whether the system can respond in the current turn, it must generate a response from the perspective of its assigned role.

ChatMDG first parses the discourse structure of multi-party dialogue and represents it as a semantic-enhanced graph, which can enhance the analysis and tracking of the dynamic role interaction based on the constructed graph. Then, the dialogue flow fused with role-interaction information can be cooperatively encoded. Additionally, due to the higher requirement for response diversity in chat-based dialogue systems, ChatMDG introduces a generative model based on deep neural networks for interaction strategy controller and response generation. Our proposed approach not only fuses the intricate graph structure inherent in multi-party dialogues but also incorporates an interaction control module, representing a significant contribution to enhancing the overall effectiveness of multi-party dialogue generation.

We propose ChatMDG, an information fusion-based multi-party dialogue generation model that incorporates discourse parsing graph networks. ChatMDG first parses the discourse structure of multi-party dialogue and represents it as a semantic-enhanced graph, which can enhance the analysis and tracking of the dynamic role interaction based on the constructed graph. Then, the dialogue flow fused with role-interaction information can be cooperatively encoded. Additionally, given the increased demand for response diversity in chat-based dialogue systems, ChatMDG introduces a generative model based on deep neural networks for interaction strategy control and response generation.

In summary, our contributions are listed as follows:

- To tackle the challenges associated with encoding and generating responses in the intricate framework of multi-party dialogues, we introduce ChatMDG, an innovative approach grounded in information fusion for multi-party dialogue generation.
- We innovatively introduce a discourse parsing graph fusion method, designed to intricately capture the interplay of role interactions within the dialogue flow, thereby significantly improving the encoding and comprehension of multi-party dialogue contexts.
- We incorporate a mediator module specifically designed for interaction control, this module ensures that conversations flow more coherently and mimic the natural dynamics of human interaction in complex multi-user scenarios.
- ChatMDG performs better on the Ubuntu IRC dataset, achieving a state-of-the-art accuracy of 79.55% in the task of next role prediction.

2. Related work

2.1. Multi-party dialogue

Multi-party dialogue, which involves multiple speakers engaging in conversation, is a challenging and significant research area. Previous studies have explored various aspects of multi-party dialogue.

Molweni first constructed a multi-party dialogue reading comprehension dataset with dialogue structural relations [7]; another work proposed a new task of multi-party chat, where conversational agents interact with humans and models in group settings, and developed a new dataset MultiLIGHT for this task [8], SDMPED introduced a new task of multi-party empathetic dialogue generation, where the goal is to generate empathetic responses for multiple speakers with different emotions [9], Shi proposed a deep sequential model for discourse parsing on multi-party dialogues [10], SSA-GNN proposed a structured perception model to analyze multi-party dialogue relations in a nonlinear way [11], Chi used the matrix tree learning algorithm to construct the correlation between utterances in multi-party dialogue [12], ThreadEncoder model encoded multi-party dialogues by dividing them into multiple linear utterances groups [13], The Structure aware sequence-to-sequence models constructed an action-behavior graph based on the basic dialogue structure graph to achieve better encoding of multi-party dialogue [14]. GroundHog focuses on dialogue generation technologies, particularly on how to effectively utilize multi-grained linguistic inputs in multi-party dialogues [15]. Addlesee discusses the construction of social robots capable of participating in multi-party dialogues using LLMs [16]. SDS explores how patients, their companions, and social robots engage in multimodal conversations within a hospital setting. This study illustrates the implementation of a multi-party multimodal dialogue system in a medical context, showcasing the potential of social robots to facilitate communication and support in real-world healthcare environments [17].

However, most of these studies do not provide a comprehensive and general framework for multi-party dialogue, and cannot handle diverse and complex situations [18]. The complexity of multi-party dialogue is compounded by the need to manage various dialogue threads, understand context shifts, and recognize the interplay of interaction cues and norms.

2.2. Dialogue generation

Dialogue generation has been a key area of research in natural language processing, aiming to create human-like conversations between machines and humans. Existing methods for Dialogue Generation can generally be divided into generation-based or retrieval-based methods. COMEDY aims to enhance traditional retrieval-based dialogue systems by better understanding dialogue history and grounding responses in past conversational context [19]. Seq2Set2Seq introduces a two-stage approach combining retrieval-based and neural generative methods to enhance response relevance and diversity, particularly in social media contexts [20]. Transformer-ED focus has been on knowledge-grounded dialogue generation, where the aim is to enhance the factual accuracy and informativeness of responses by grounding them in external knowledge sources [21]. Furthermore, IKA introduces a plug-and-play retrieval-based framework designed to enhance LLMs for knowledge-grounded dialogue generation, emphasizing the importance of in-context learning [22]. A notable limitation inherent in retrieval-based methodologies is their reliance on a pre-established repository of potential utterances. This dependence often culminates in a constrained diversity of generated responses, potentially stifling the dynamism and spontaneity characteristic of human dialogues.

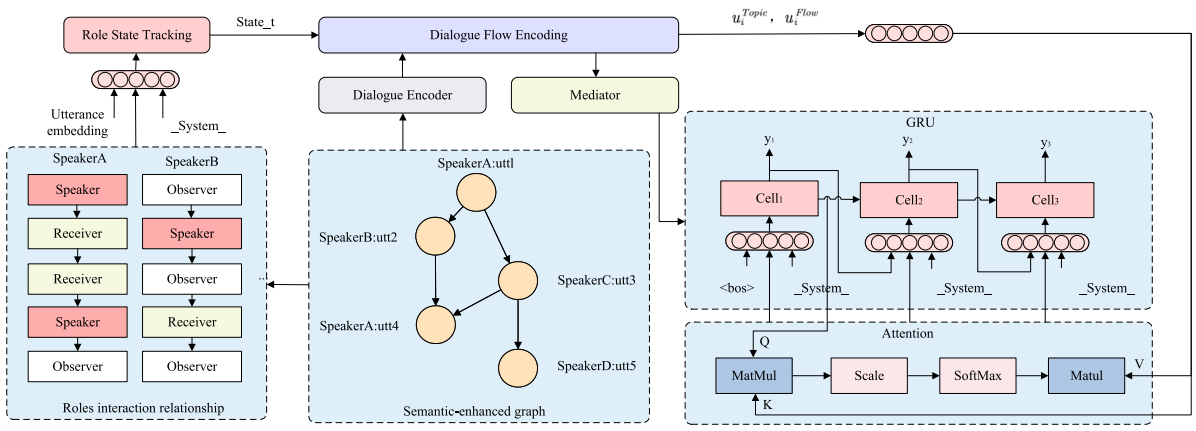

Fig. 2. Architecture of ChatMDG. **Discourse Parsing**: Parses the structural and thematic facets of multi-party dialogues as a semantic-enriched graph; **Role Interaction Analyzing**: Analyzes how different roles interact with each other, considering their dependencies and dynamics during the conversation; **Role State Tracking**: Dynamically monitors and captures the current state of each role based on their interaction features; **Dialogue Flow Encoding**: Encodes the overall context and dynamics of the conversation by incorporating the collective features of all roles; **Interaction Mediating**: Controls the interaction mode within the dialogue to facilitate smooth communication between roles; **Response Generating**: Generates appropriate and contextually relevant response utterances based on the encoded dialogue flow and guidance from the mediator.

In contrast, generative models based on deep learning can effectively solve these problems. Iulian and Alessandro used a hierarchical encoding model to encode individual utterances and the overall dialogue. Based on this method [23], HRAN combined word-level attention mechanism and utterance-level attention mechanism in response generation [24]. Another typical work strengthened the quality of dialogue generation by using static and dynamic attention mechanisms [25]. DIALOGPT adapts the GPT-2 [26] architecture to learn conversational patterns from a massive dataset of human-human dialogue exchanges, enabling it to generate human-like responses in a wide range of conversational contexts [27]. A knowledge-augmented stylized dialogue generation model aims to produce coherent and context-aware dialogues that effectively emulate the desired style, making significant contributions to the field of stylized dialogue [28]. Also, the role of large pre-trained models in dialogue generation has been a significant area of interest. These models have been adapted and fine-tuned for specific dialogue generation tasks, leveraging their vast knowledge bases and generative capabilities [29].

For multi-party dialogue generation research, there have been retrieval-based methods in the past. [30] proposed a dynamic representation method for role state representation. [31] proposed a speaker interaction RNN to model dialogue role interactions. [32] used multi-task learning to perform dynamic topic tracking and response selection tasks in dialogue. In generation-based methods, [33] modeled dialogue role states based on explicit reply relationships in group chat data. [34] proposed a structured model for multi-party dialogue generation. [35] then merged structured attention mechanisms into a variational recurrent neural network. [36] proposed an attention mechanism for dialogue receivers based on the Transformer architecture for multi-party dialogue generation. Another study explored the development of a multi-party conversational social robot powered by LLMs. This work aimed to enhance the interaction capabilities of robots in group settings, making them more adaptable to multi-party dialogues [37]. Overall, The advancements in this field are driven by the need to better understand and manage the intricacies of conversations involving multiple participants, making AI dialogue systems more adept at navigating the complexities of human communication.

3. Problem formulation

A multi-party conversation, characterized by the involvement of more than two participants, presents a complex and applied scenario necessitating a comprehensive grasp of the contextual backdrop, identities of the speakers and addressees, as well as the thematic elements underpinning the dialogue. One possible way to formulate the multi-party conversation problem is to use a dependency model that represents the relations between utterances and speakers in a conversation graph. In ReDE [38], An approach given a sequence of utterances $U = \{u_0, u_1, \ldots, u_n\}$, each spoken by a speaker $s(u_i)$, then a function $I(i)$ that assigns the $i_t h$ token to its utterance can be defined. Then, a dependency parser can be trained to predict the parent utterance and speaker for each utterance in U based on the features of the tokens and speakers. Another possible way to formulate the multi-party conversation problem is to use a dynamic topic-tracking model that selects the best response for a given context based on the topic similarity between the context and the response. In Topic-BERT [32], given a set of candidate responses $R = \{r_0, r_1, \ldots, r_n\}$ for a context $C = \{c_0, c_1, \ldots, c_n\}$, each utterance in C and R can be encoded using a pre-trained language model like BERT. Then, the topic similarity between each pair of utterances in C and R can be computed using a topic classifier. Finally, the response that has the highest average topic similarity with the prevailing context is designated for selection.

The generation of multi-party dialogues represents a sophisticated challenge, aimed at empowering dialogue systems to partake in intricate conversations featuring a diversity of roles and a series of conversational exchanges. Contrary to traditional human–machine interactions, where the system alternates with a single user, multi-party dialogues require a different interaction mode that adapts to the characteristics of multi-role interactions. In particular, the system must ascertain the necessity of a response, grounded in the pre-existing dialogue trajectory, and contingent upon such a determination, craft a response that is not only contextually pertinent but also aligns with the specific role it embodies within the conversation. This task can be formulated as follows:

$$\mathrm{f}_{\mathrm{interpose}} = \begin{cases} 1 & \text{SYSTEM in roles}_{\mathrm{pred}} \\ 0 & \text{SYSTEM not in roles}_{\mathrm{pred}} \end{cases} \tag{1}$$

$$\bar{r} = \arg\max_{r} \log\ P(r \mid G, S) = \arg\max_{r} \sum_{k=1}^{|r|} \log\ P\left(r_k \mid G_{<k}, S\right) \tag{2}$$

where $f_{interpose}$ determines the necessity of the system's response in the given dialogue turn, $roles_{pred}$ denotes the set of roles that are suitable for dialogue response in the current turn, G encapsulates the dialogue history in the current turn, and S represents the speaker, which is the system role. The response r is formulated autoregressively and is manifested solely when $f_{interpose}$ equals 1.

In addressing this challenge, we introduce a novel generative model-based neural network technology, that leverages graph structures to model the reply relationships between multi-party dialogue utterances. The possibility of multiple roles being suitable for dialogue response within the same turn is also considered, and the interaction mode is controlled through active and passive modes. Additionally, the distinct characteristics associated with the speaker's role are meticulously factored into the response generation process.

4. Multi-party dialogue generative model

In this study, we introduce ChatMDG, a neural network-based model designed for generating multi-party dialogues, which harnesses the discourse structure graph for enhanced dialogue generation. The architecture of ChatMDG comprises an encoder and a decoder, with the model's structure depicted in Fig. 2. The encoding module ingests the discourse structure graph, monitoring the distinctive identity features of each participant within the dialogue. It subsequently generates state vectors corresponding to each role, amalgamating these vectors to encapsulate the dynamics of the dialogue flow. The decoding module, predicated on the current state, adjudicates the appropriateness of response generation at a given turn. Ultimately, the model synthesizes a response utterance congruent with the perspective of each involved role.

4.1. Discourse parsing module

Utterance Encoder. For a multi-party dialogue D comprising n rounds and m distinct roles, the notation can be expressed as follows:

$$D = \left\{ \left(u_1, s_1\right), \left(u_2, s_2\right), \ldots, \left(u_n, s_n\right) \right\} \tag{3}$$

$$S\left(D\right) = \left\{ s_1, s_2, \ldots, s_m \right\} \tag{4}$$

where utt_i represents the utterance in the i_{th} round, spk_j denotes the label of the j_{th} role, and the pair (utt_i, spk_j) signifies the utterance made by role j in round i. The notation $S(D)$ defines the set encompassing all roles present within dialogue D.

To encode each utterance utt_i, an encoding layer is used first to obtain a contextualized word representation for every word within the utterance, utilizing BERT [39] for this purpose. Subsequently, a bidirectional GRU layer is applied to learn a comprehensive vector representation of the entire utterance. The last hidden states from both directions are concatenated to yield the final utterance representation h_i^u.

Semantic-enriched Graph. In order to improve the model's comprehension of multi-party dialogues, we propose a method that involves analyzing the discourse structure of the dialogue, dissecting the intricate interplays between utterances and roles, and formulating a discourse structure graph. a Graph Attention Network (GAT) [40] is utilized, facilitating a nuanced understanding of the dialogue's architectural dynamics.

Initially, the encoding of the context utterances is computed. For the i_{th} round of roles, the static embedding vector e_i is derived from an embedding matrix, Subsequently, a bidirectional GRU network is employed to acquire the corresponding role vector h_i^s for each round of dialogue round, formulated as $h_i^s = BiGRU\left(e_i\right)$.

In this setup, the bidirectional GRU network concatenates outputs contingent upon their positional context and amalgamates them with utterance vectors to derive the encoding of the utterance unit, represented as $u_i = \left[h_i^u; h_i^s\right]$. To achieve the context encoding of the utterance u_i, multiple layers of a self-attention mechanism are applied, formulated as $u_i = Self\,Attention\left(h_i\right)$.

Then the input graph for the GAT is established, adopting the structure of a fully connected graph. each utterance is designated as a node, with the initial node representation v_i^0 equated to h_i. Regarding the relational edges between nodes, two features are taken into account: r_{ij}^s denotes whether utterance i and utterance j originate from the identical speaker, and r_{ij}^d signifies the relative positional distance between utterance i and utterance j. The initial representation of an edge is articulated as $r_{ij}^0 = \left[r_{ij}^s; r_{ij}^d\right]$.

In the l_{th} layer of the graph network, the representation of each node is denoted as v_i^l, while the representation of each relationship edge in the l_{th} layer is expressed as r_{ij}^l. In line with the methodology proposed by Yang et al. [41], the operational mechanism of the GAT is divided into principal phases: node feature aggregation and edge feature aggregation. For each node, attention vectors pertaining to its adjacent nodes and edges are computed independently and subsequently synthesized via a mapping matrix, culminating in the derivation of the node feature vector v_i^l post the l_{th} layer within the graph attention network. Through the intersection points of the adjacent edges, the feature information is disseminated, facilitating the computation of the edge feature vector r_{ij}^l subsequent to the l_{th} layer. The node feature is computed as follows:

$$v_i^l = W_l^v[\sum_{v_j \in V(v_i)} \alpha_{ij}^v v_j^{l-1}; \sum_{r_i j \in E(v_i)} \alpha_{ij}^r r_{ij}^{l-1}] \tag{5}$$

The edge feature is determined by:

$$r_{ij}^l = W_l^r[r_{ij}^{l-1}; \alpha_i^{edge} v_i^{l-1} + \alpha_j^{edge} v_j^{l-1}] \tag{6}$$

where W_l^v and W_l^r represent the trainable parameter matrices for the node and edge features respectively, α_{ij}^v and α_{ij}^r denote the attention weights assigned to the current node in relation to its neighboring nodes and edges. $V(\cdot)$ and $E(\cdot)$ correspond to the sets of neighboring nodes and edges adjacent to the current node. In the context of each relational edge, nodes i and j are identified as the nodes linked by the edge, with α_i^{edge} and α_j^{edge} representing the attention weights allocated to these nodes respectively.

Following L iterations, the concluding feature vector for any given relational edge is acquired as r_{ij}^L, with the multi-party dialogue structure conceptualized as a graph. Subsequently, a sigmoid function is employed to ascertain the relevance of each relational edge:

$$score_{ij} = sigmoid(W^S[r_{ij}^L; r_{ji}^L]) \tag{7}$$

Where W^s denotes a learnable parameter matrix. a predefined threshold is applied to exclude edges of low relevance, thereby constructing the discourse structure graph.

4.2. Dialogue encoding module

Role State Tracking. Within the framework of multi-party dialogues, utterances unfold sequentially across numerous rounds. Each round is initiated by a speaker delivering an utterance. Following Liu et al. [33], the state of each speaker is updated by analyzing their identity characteristics. The roles within the dialogue are classified into three types: speaker, receiver, and observer. The 'speaker' refers to the entity responsible for generating each utterance, with the stipulation of a singular speaker per round. The 'receiver' is designated as the intended recipient of each utterance, with the role associated with the antecedent utterance within the discourse structure graph deemed the receiver for the current utterance, acknowledging the possibility of multiple receivers. Conversely, the 'observer' encompasses any role that does not fulfill the criteria of being either the speaker or the receiver in the dialogue context.

To encapsulate the feature propagation pertaining to the three roles of speaker, receiver, and observer within the dialogue, three distinct networks are established: $GRU^{Speaker}$, $GRU^{Receiver}$ and $GRU^{Observer}$. The characteristics of both the speaker's target and the current utterance's receiver are taken into consideration. For example, in the process of updating the speaker's feature, the receiver's feature is integrated, given that the receiver's role provides insight into the response to the utterance. The state of each role is meticulously tracked through the

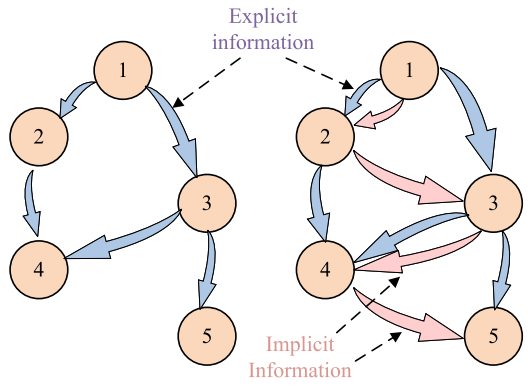

Fig. 3. Multi-party dialogue topic linking. Local dialogue flow is defined as a graph created by linking the utterance of the current turn to its parent node. Topic information comprises a sequential aggregation of all utterances, encapsulating the central theme of the current multi-party dialogue.

utterances of each round, with h_t^u representing the vector representation of the utterance in the t_{th} round. The dynamic role state s_i^t of role i during the t_{th} round of the dialogue is calculated as follows:

$$s_i^t = \begin{cases} GRU^{Speaker}\left(h_t^u, \left[s_i^{t-1}; s_{receiver}^{t-1}\right]\right) \\ role^i \ is \ speaker \\ GRU^{Receiver}\left(h_t^u, \left[s_i^{t-1}; s_{speaker}^{t-1}\right]\right) \\ role^i \ is \ receiver \\ GRU^{Observer}\left(h_t^u, \left[s_i^{t-1}; s_{speaker}^{t-1}; s_{receiver}^{t-1}\right]\right) \\ role^i \ is \ observer \end{cases} \tag{8}$$

Where $s_{speaker}^{t-1}$ represents the state vector of the speaker role from the $t-1$ round, which is assigned the same value due to the presence of a singular speaker per round. The term $s_{receiver}^{t-1}$ signifies the cumulative state of the receiver role in the $t-1$ round. Since there may be any number of receivers in each round, their respective vectors are averaged to derive this composite state vector.

Graph Fusion Encoder. Multi-party dialogue usually involves multiple sub-dialogue flows, thereby bifurcating dialogue features into two distinct segments: local dialogue flow information and dialogue topic information. The local dialogue flow pertains to the dialogue graph created by linking the utterances of the current round with their parent node, manifesting as a divergent branch within the dialogue's evolution and encapsulating core-related information. The dialogue topic represents the current theme of the multi-party dialogue. Despite the intricate branching structure inherent in multi-party dialogues, it often focuses on a theme. As shown in Fig. 3, the dialogue topic is linked chronologically to convey the thematic features of the dialogue.

For each utterance, distinct vector representations are derived $\left\{h_1^u, h_2^u, \ldots, h_n^u\right\}$. Subsequently, $Spk(i)$ signifies the dialogue role for the ith dialogue round, and the utterance representation is amalgamated with the role embedding $s_{Spk(i)}^i$ to form a novel representation for each utterance unit $h_1^u = \left[h_n^u; s_{Spk(i)}^i\right]$. For local dialogue flow, the spatial structural information is represented by the discourse structure graph, and the utterance feature is enhanced with the features of its spatial neighbors. As illustrated in Fig. 3, Utterance 2 and Utterance 3 are two different responses to Utterance 1, and both utterances can help to represent Utterance 1. Meanwhile, Utterances 2 and 3 may have parallel semantic relationships. Therefore, as shown in Fig. 4, a neighborhood attention mechanism is employed to secure the vector representation $\tilde{h}_i$ of the utterance.

Next, the temporal development process of the dialogue flow is considered. Since the dialogue flow has a directed acyclic graph structure, each utterance is capable of simultaneously having an indefinite

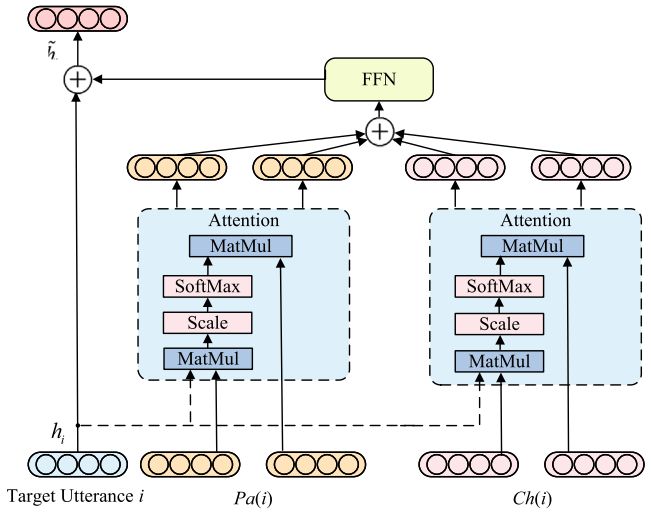

Fig. 4. Spatial Structural Attention Mechanism. This architecture delineates the update process for the target utterance i. The symbols $Ch(i)$ and $Pa(i)$ signify the collections of sub-utterances (child utterances) and parent utterances.

number of parent and child utterances. Therefore, the DAG-GRU network [42] is adopted to propagate the information of the dialogue flow. A deep post-fusion function $g(\cdot)$ is applied to extract the local dialogue flow information u_i^{Flow} corresponding to the ith round of the dialogue:

$$u_i^{Flow} = g\left(\left\{\tilde{u}_j \,|\, j \in Pa(i)\right\}\right) \tag{9}$$

$$\tilde{u}_j = GRU\left(\tilde{h}_i, u_j\right), j \in Pa(i) \tag{10}$$

Where $g(\cdot)$ represents the fusion function, and every edge within the discourse structure graph is assigned a substantial weight $score_{ij}$, reflecting the significance of the utterance in relation to its parent utterance.

To equip the dialogue generative model with the capability to discern and select task-relevant features from a variety of parent utterances, a fusion function is implemented by amalgamating a static weight with a dynamic gate weight:

$$g\left(\left\{\tilde{u}_j \,|\, j \in Pa(i)\right\}\right) = LayerNorm\left(\sum_{j \in Pa(i)} score_{ij}\tilde{u}_j + \sum_{j \in Pa(i)} \alpha_{ij}\tilde{u}_j\right) \tag{11}$$

$$\alpha_{ij} = \frac{\sigma\left(W^g\left[\tilde{h}_i; \tilde{u}_j\right] + b^g\right)}{\sum_{k \in Pa(i)} \sigma\left(W^g\left[\tilde{h}_i; \tilde{u}_k\right] + b^g\right)} \tag{12}$$

where W^g and b^g are the gate mechanism parameters.

To capture and track the dialogue topic features, the dialogue sequence is separately encoded to obtain the multi-party dialogue topic feature vector u_i^{Topic}:

$$u_i^{Topic} = GRU\left(\tilde{h}_i, u_{i-1}^{Topic}\right) \tag{13}$$

Then, the local dialogue flow vector and the dialogue topic vector are concatenated to yield the utterance representation $u_i = \left[u_i^{Topic}; u_i^{Flow}\right]$ for the ith turn in the dialogue.

4.3. Dialogue decoding module

Dialogue features can be categorized into two main components: local dialogue flow information and dialogue topic information. The local dialogue flow pertains to the dialogue graph created by linking the utterances of the current round to their parent node, representing a developmental branch of the dialogue that encapsulates pivotal information. On the other hand, the dialogue topic encapsulates the

prevailing thematic essence of the multi-party dialogue. Despite the intricate branching architecture characteristic of multi-party dialogues, they generally center around a dominant theme.

Distinct vector representations for each utterance are derived, denoted as $\{h_1^u, h_2^u, \ldots, h_n^u\}$. Subsequently, $Spk(i)$ signifies the dialogue role for the ith round, and the utterance representation is amalgamated with the corresponding role embedding $s^i_{Spk(i)}$ to formulate a novel representation for each utterance unit, expressed as $h_1^u = [h_n^u; s^i_{Spk(i)}]$. In terms of local dialogue flow, the spatial structural information is represented by the discourse structure graph, and the utterance feature is enhanced with the features of its spatial neighbors.

A neighborhood attention mechanism is used to obtain the vector representation $\tilde{h}_i$ of the utterance:

$$g\left(\{\tilde{u}_j \,|j \in Pa(i)\}\right) = LN\Big(\sum_{j\in Pa(i)} s_{ij}\tilde{u}_j + \sum_{j\in Pa(i)} \alpha_{ij}\tilde{u}_j\Big) \tag{14}$$

Where LN denotes Layer Normalization, $Ch(i)$ represents the collection of sub-utterances linked to utterance i, and $Pa(i)$ indicates the set of parent utterances connected to utterance i, The attention weights α^c_{ij} and α^p_{ij} are calculated for the current utterance in relation to its adjacent sub-utterances and parent utterances, respectively.

Next, the temporal development process of the dialogue flow is considered. The DAG-GRU network [42] is adopted to propagate the information of the dialogue flow. A deep post-fusion function $g(\cdot)$ is utilized to extract the local dialogue flow information u_i^{Flow} at the ith round of dialogue.

To equip the dialogue generative model with the capability to discern and select task-relevant features from a diverse array of parent utterances, a fusion function is implemented by integrating a static weight with a dynamic gate weight:

$$\alpha_{ij} = \frac{\sigma(W^g[\tilde{h}_i;\tilde{u}_j] + b^g)}{\sum_{k\in Pa(i)} \sigma(W^g[\tilde{h}_i;\tilde{u}_k] + b^g)} \tag{15}$$

where W^g and b^g are the gate mechanism parameters.

To effectively capture and monitor the features of the dialogue topic, the dialogue sequence is encoded independently, yielding the multi-party dialogue topic feature vector u_i^{Topic}:

$$u_i^{Topic} = GRU(\tilde{h}_i, u_{i-1}^{Topic}) \tag{16}$$

Subsequently, the local dialogue flow vector and the dialogue topic vector are concatenated to generate the utterance representation $u_i = [u_i^{Topic}; u_i^{Flow}]$ for the ith turn in the dialogue.

Interaction Mediator. This module addresses the interaction challenges encountered by the system in multi-party dialogue generation. As the system assumes a role in multi-party dialogues, it must determine the appropriate moments to engage. A mediator module is thus implemented to manage the system's mode of interaction. The scenarios for dialogue response are bifurcated: one scenario involves other roles initiating interaction with the system, eliciting a passive response from the system; the alternate scenario arises when the system takes an interest in the ongoing topic among other roles, prompting it to actively participate and contribute to the dialogue.

The passive mode pertains to a user-centric approach in the human–computer dialogue process, wherein the user governs both the commencement and conclusion of the dialogue. In this mode, the system reacts to each user's utterance sequentially until the user ceases to respond, effectively ending the dialogue flow. Leveraging the discourse structure graph, which links the current utterance to its antecedent, if the speaker attributed to a parent utterance assumes the system's role, the ensuing utterance is interpreted as being addressed to the system. Consequently, it necessitates a response from the system side.

In active mode, the system is required to proactively engage in the current dialogue. Owing to the distinct nature of interaction in multi-party dialogues, the system must ascertain the opportune moments to integrate itself into the current dialogue flow. Supervised learning techniques are employed to forecast the subsequent speaker within the dialogue. When it is anticipated that the system itself is to assume the next role in the dialogue, a response utterance is crafted. The prediction of the forthcoming dialogue role, denoted as spk_{reply}, is calculated in the following manner:

$$apk_{reply} = arg\max(\alpha) \tag{17}$$

$$\alpha = \frac{v^T \tanh\left(W^g u_n + U^g s_e^n\right)}{\sum_{p\in S(D)} v^T \tanh\left(W^g u_n + U^g s_p^n\right)} \tag{18}$$

Where W^g and U^g represent the learnable parameter matrices, u_n signifies the state code of the dialogue flow obtained for the current round, and s_e^n indicates the state code of the dialogue role e in the current round.

Response Generation. Upon the Interaction Mediator module ascertaining the necessity for the system side to formulate a response in the current turn, it produces a response. In the context of multi-party dialogue, it is imperative to take into account the role information to ensure that the generated response is congruent with the stance of the pertinent speaker role. Therefore, a GRU module is employed to implement the decoder, with the incorporation of an attention mechanism to elevate the response quality. The generation sequence for the tth word y_t within the response unfolds as follows:

$$y_t = arg\max\left(p_t\right) \tag{19}$$

$$p_t = softmax\left(W^o\left[c_t; s^n_{SYSTEM}; f_t\right]\right) \tag{20}$$

$$f_t = GRU\left(f_{t-1}, \left[c_t; s^n_{SYSTEM}; Emb\left(y_{T-1}\right)\right]\right) \tag{21}$$

Where W^o represents a trainable parameter matrix, $Emb(\cdot)$ signifies the word embedding operation, and the initial state f_0 of the GRU is initialized using the current dialogue flow state vector u_n. The term s^n_{SYSTEM} refers to the current role state vector of the system role within the dialogue.

The utterance-level attention vector c_t at the tth time step is computed as follows:

$$c_t = \sum_{i=1}^{n} \alpha_i \tilde{h}_i \tag{22}$$

$$\alpha_i = \frac{exp\left(f_{t-1}^T W^a \tilde{h}_i\right)}{\sum_{k=1}^{n} \exp\left(f_{t-1}^T W^a \tilde{h}_k\right)} \tag{23}$$

The W^a represents a learnable parameter matrix and $\tilde{h}_i$ signifies the deep vector representation of the utterance, which is derived through the neighborhood attention mechanism. Ultimately, to achieve maximal coherence and contextual relevance, the system response utterance is generated through the application of the Beam Search algorithm.

5. Experiments

5.1. Experimental setup

Datasets. The Ubuntu IRC dataset [43] is used and divided into multiple groups of dialogues based on window size. The dialogues are further filtered by limiting the maximum length of the utterances, removing dialogues with excessive punctuation and those that are too short, and eliminating dialogues with only two roles. Finally, 91364 groups of multi-party dialogues are randomly sampled for training, validation, and testing, with 85364/3000/3000 groups of dialogues being used for training, validation, and testing respectively.

Training details. The BERT model with bert-base-cased embedding is employed for embedding the input textual utterances, with the final two layers subjected to fine-tuning during the training phase. The dimensions of the hidden layer vectors and role vectors are set to 300, and the number of attention heads is set to 4. Structurally, the model incorporates a dual-layer GRU architecture. The Adam optimizer

Table 1
The next role prediction accuracy of the active module in the Interaction Mediator, which is crucial for facilitating seamless transitions and preserving the coherence of multi-party dialogues.

Model	Accuracy(%)	Time(ms)
Static-ADR	52.9	245
Dynamic-ADR	59.0	267
ChatMDG-Seq	61.8	294
ChatMDG-Tree	73.2	301
ChatMDG-DAG	**79.5**	312

Table 2
The experimental evaluation of the Interaction Mediator in three different modes.

Mode	Interaction mediator		
	Precision (%)	Recall (%)	F1 (%)
Active	**85.9**	62.7	72.5
Passive	73.6	82.0	77.6
Mixed	76.3	**90.6**	**82.8**

orchestrates the training regimen, initiating with a learning rate of 5e-4, which is progressively attenuated in tandem with the training evolution. Configurational parameters include a batch size of 16, a dropout rate of 0.3, and an upper limit of 30 for utterance length.

Evaluation metrics. Following the evaluation metrics for multi-party dialogue systems summarized by Mahajan et al. [44], multiple evaluation metrics are used to measure different aspects of the system's performance. Precision, recall, and F1 scores are used to evaluate the effectiveness of the I Interaction Mediator module, while the quality of dialogue generation is evaluated using BLEU-1, BLEU-2, BLEU, ROUGE, and METEOR metrics, all sourced from the NLTK toolkit.

5.2. Results and analysis

Interaction Mediator evaluation. The Interaction Mediator module's active inclusion mode is operationalized by forecasting the subsequent dialogue role. Our model is benchmarked against Static-ADR [30] and Dynamic-ADR [30]. Our Model-Seq uses a linear connection to the dialogue history, so it cannot determine the recipient's identity. Our Model-Tree associates each dialogue utterance with only one parent utterance, while Our Model-DAG uses a graph structure, where each utterance can have any number of associated parent utterances. The empirical findings presented in Table 1 elucidate that Static-ADR merely leverages embedding matrices for role encoding in a topological sequence, thus failing to amalgamate role-specific information within the dialogue effectively. In contrast, other models amalgamate utterance data to refresh the role vector, indicating that the inclusion of utterance data aids in depicting role states more accurately. A comparative analysis of sequence, tree, and graph-based models reveals the beneficial influence of dissecting the dialogue's discourse structure and refining the roles' identity attributes. The graph-based model exhibits superior role feature learning capabilities, underscoring the efficacy of the graph structure for reply relationships derived from discourse parsing in this study over the tree-structured approach. This results in more authentic role identity predictions and heightened accuracy in prognosticating the forthcoming speaking role.

The overall testing is conducted on the dialogue interaction control module to assess its functionality in authentic multi-party dialogue scenarios, where multiple roles may concurrently be apt for a response. To this end, dialogues are partitioned into two segments at a 2:1 ratio, with the initial segment constituting the dialogue history and the latter segment identifying all roles eligible to respond to the concluding utterance in the dialogue history. The model's capability to ascertain the suitability of each role for response under varying interaction modes is considered. The results depicted in Table 2 indicate that the hybrid mode, which merges passive and active modes, enhances the system's response propensity, resulting in the highest recall rate among the three modes, with precision falling between the other two. By integrating the active and passive modes, the hybrid mode effectively meets the system's requirement to maintain the dialogue flow while also engaging actively in the multi-party dialogue.

Response generation evaluation. To validate the effectiveness of our proposed model in multi-party dialogue scenarios, it is benchmarked against a selection of seminal methodologies within the field of generative dialogue, including the following baseline methods: (1) Seq2Seq [45], A foundational sequence-to-sequence architecture that integrates dialogue history for enhanced understanding. (2) HRED [23], Utilizes a hierarchical encoder to encode utterances and dialogues separately, capturing the nuanced layers of dialogue structure. (3) HRAN [46], Employs a dual attention mechanism during decoding to concentrate on each utterance and word within the dialogue history. (4) HSAN [47], Leverages a hierarchical self-attention mechanism to deepen dialogue comprehension and facilitate generation. (5) HSAN-spk [47], Adopts the same role-state tracking module as our model to delineate various dialogue roles distinctly. (6) ICRED [33], Models individual dialogue role characteristics in multi-party settings and maintains a contextual representation of each role's latest utterance through a memory layer. (7) GSN [34], Introduces a graph-based encoding framework that distinguishes speakers but cannot model inter-speaker relationships, connecting utterances from the same speaker with hidden edges. (8) ASRG [36], Also captures the state of each dialogue role and employs an attention mechanism to discern the intended recipient of the dialogue during the generation phase. These methodologies provide a comprehensive spectrum of approaches for understanding and generating multi-party dialogues, against which our model's performance is meticulously assessed.

As shown in Table 3, our proposed method outperforms all the baseline methods in multi-party dialogue scenarios. The findings reveal that models such as Seq2Seq, HRED, HRAN, and HSAN, which neglect the roles within the dialogue and its structural aspects, exhibit inferior performance compared to models specifically designed for multi-party dialogue generation. HSAN-spk and ICRED, which discern between different dialogue roles, show enhanced outcomes yet still perceive dialogues as linear sequences, overlooking the structured nature inherent to multi-party dialogues. On the other hand, GSN and ASRG, which are predicated on tree and hierarchical structures respectively, manifest a degree of improvement. However, GSN falls short in explicitly modeling the state of dialogue roles, and ASRG does not comprehensively leverage the prior structural features of multi-party dialogues. Our ChatMDG, by incorporating the prior graph structural attributes of multi-party dialogues and dynamically representing each role's state, facilitates a more precise learning of dialogue semantics and the stance information pertinent to each role. Compared with Our Model-Tree and Our Model-Seq, our graph-based ChatMDG achieves state-of-the-art performance, demonstrating the effectiveness and modeling multi-party dialogue through graph structures and methodology.

Furthermore, to substantiate the efficacy of our proposed graph-based framework, we conducted a comprehensive evaluation of its performance under diverse configurations of dialogue turns and roles, as illustrated in Fig. 5. The results indicate that our graph-based multi-party dialogue modeling method achieves relatively better performance in more complex multi-party dialogue structures. This observation is mainly because when the number of dialogue turns and roles are small, the dialogue structure is relatively simple, rendering elementary linear modeling techniques partially effective. However, with the escalation in the number of dialogue turns and roles, the complexity of the dialogue structure intensifies, and the linear modeling method cannot accurately understand the development process of the dialogue or capture cross-turn correlated dialogue utterance information. Conversely, our graph-based methodology distinctly constructs the dialogue's graph architecture, thereby enabling the model to intricately trace and encode the developmental dynamics of the dialogue flow. This facilitates a precise depiction of each role's state, culminating in a demonstrably stable performance in complex multi-party dialogue scenarios.

Table 3
Response generation performance (%) of ChatMDG and baseline approaches on the UbuntuIRC dataset.

Model	UbuntuIRC				
	BLEU-1	BLEU-2	BLEU	ROUGE	METEOR
Seq2Seq [45]	5.9714	1.7804	0.5309	0.5287	3.7833
HRED [23]	6.3436	1.9116	0.5945	0.5861	4.1009
HRAN [46]	6.5532	2.0540	0.6194	0.6117	4.2842
HSAN [47]	7.7067	2.1458	0.6743	0.6689	4.8170
ICRED [33]	8.2895	2.5116	0.7232	0.7161	5.3526
HSAN-spk [47]	8.9728	2.7466	0.8018	0.7590	5.8271
GSN [34]	9.0254	2.7821	0.7837	0.7728	5.9185
ASRG [36]	9.4082	2.8896	0.8621	0.8509	6.5987
ChatMDG-seq	7.9723	2.2853	0.7081	0.6844	5.1534
ChatMDG-tree	9.3951	2.7687	0.8588	0.8361	6.3623
ChatMDG (Ours)	**9.7242**	**3.0416**	**0.9064**	**0.8903**	**6.8961**

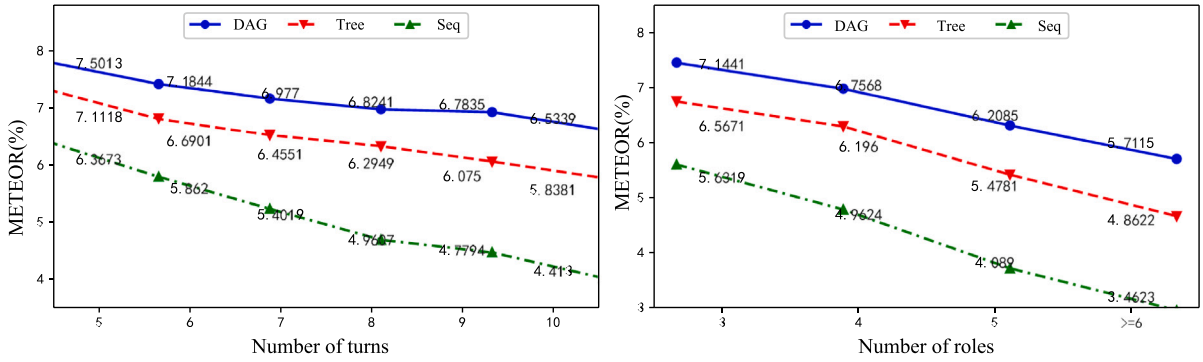

Fig. 5. The Impact of Different Turns and Roles on Different Discourse Structures. A Comparative Analysis of semantic-enhanced graphs based on Directed Acyclic Graph (DAG), Tree, and Sequence.

Table 4
Examples of Case study on Interactive Mediator.

Turn	Role	Utterance
1	SpeakerA	How can I install new packages into Ubuntu?
2	**SYS**	**You** can install the package from the repositories. (Respond to SpeakerA)
3	SpeakerB	A few ways, you can use synaptic or apt-get.
4	SpeakerC	You need to use it with sudo though.
5	SpeakerA	But it is not working.
6	**SYS**	What do **you** mean it does not work? (Respond to SpeakerA)
7	SpeakerC	What error messages are you getting? That might give you a clue on how to fix it.
8	SpeakerA	It is saying that the package has broken dependencies.
9	**SYS**	**You** can try to fix broken dependencies before install the package. (Respond to SpeakerA)
10	SpeakerB	Yes, using the command sudo apt-get install -f to fix it.
11	**SYS**	It is a workaround as well, though. (Respond to SpeakerB)

5.3. Case study

Case study on Interactive Mediator. In multi-party dialogue generation, it is imperative for the system to adeptly manage its mode of interaction. To this end, we initially orchestrated a simulated dialogue encompassing the system and multiple users, grounded in real-world scenarios, with the empirical outcomes presented in Table 4. Within this simulated environment, *SpeakerA*, *SpeakerB*, and *SpeakerC* are three user roles controlled by manual input during the dialogue, and the *SYS* is the system role, which automatically determines whether to reply in each turn and generates a response at an appropriate time. The results detailed in Table 4 illustrate the system's capacity to judiciously navigate its dialogue interaction mode, opting to engage with utterances specifically pertinent to its designated role rather than indiscriminately responding to every message from other participants. Additionally, the utterances produced by the system not only align with its designated role perspective but also demonstrate an astute consideration of the preceding dialogue's context. For instance, in the ninth turn of the dialogue, the *SYS* generated the phrase *"You can try to fix broken dependencies before installing the package."* which reflects its understanding of the history of multi-party dialogue and takes into account the "install" issue discussed in the first and second turns.

Case study on Response generation. To validate the model's proficiency in seamlessly integrating information pertaining to role stances, it was manipulated to formulate responses while assuming various roles within a consistent historical dialogue backdrop, thereby facilitating the examination of response variations attributable to distinct role stances. Illustrated in Table 5, the first four turns are the known dialogue history, and the four utterances in the fifth turn are the utterances generated by the model as four different roles. In the first four turns of the dialogue history, *Role A's* stance is that of a questioner, while *Role B* and *Role C's* stances tend to be answerers. Based on the four different responses generated in the fifth turn, It can be seen that *Role A's* stance expresses more doubts and questions about the problem, while *Role B* and *Role C's* stances respond to *Role A's* question from different perspectives. Role B offers a solution to the problem, while *Role C* clarifies the problem further. This phenomenon is consistent with the original dialogue stance of each role. Finally, the model was also tasked with generating responses in the capacity of an observer, denoted as *Role D*. The responses thus produced were in harmony with the observer's neutral stance, exhibiting no discordance with its predefined identity, even when addressing identical issues.

6. Conclusion

The complexity of multi-party dialogues is amplified by the dynamic interplay of interactions across various roles and the multiple possibilities of the multi-party dialogue flows. In this study, we introduce

Table 5
Examples of System Response generation at different role positions.

Turn	Role	Utterance
1	SpeakerA	What webcam should I get so that it would work well in Ubuntu?
2	SpeakerB	None, webcam support is pretty universally bad. you might find some with recent support, though.
3	SpeakerC	For lists of supported hardware on ubuntu see URL – to help debugging and improving hardware detection.
4	SpeakerA	That is not too good, I am working on filepath and i need to test it with something.
5	**Role A**	I have no idea. I don't see it in offtopic.
	Role B	I think you are asking around for the support question, check your filepath.
	Role C	What kind of webcam are you using?
	Role D	I have a similar problem with the webcam from my network manager.

ChatMDG, a semantic-enhanced graph fusion methodology designed to model the complexities of multi-party dialogues, wherein each graph node is representative of an individual utterance. ChatMDG achieves discourse parsing and fuses role-interaction information for multi-party context understanding, which further instructs response generation. This strategy empowers our model with a deeper understanding of the dialogue structure across a range of interactions. The experimental results show that our proposed ChatMDG model achieves 79.55% accuracy on the Ubuntu IRC dataset, outperforming other existing models, demonstrating ChatMDG's capability to effectively navigate multi-party dialogue scenarios.

In our future work, we aim to study the utilization of the semantic-enhanced graph for the integration and regulation of LLMs, Through the fusion of graph and language synchronous generation methods, devising more interpretable and human-like interaction in the complex multi-party dialogue generation.

CRediT authorship contribution statement

Jingyang Li: Writing – original draft, Software, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Shengli Song:** Writing – review & editing, Supervision, Resources, Methodology, Conceptualization. **Yixin Li:** Writing – original draft, Visualization, Validation, Investigation. **Hanxiao Zhang:** Visualization, Validation, Software, Formal analysis, Data curation. **Guangneng Hu:** Writing – review & editing, Supervision, Resources, Methodology, Funding acquisition.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Guangneng Hu reports financial support was provided by National Natural Science Foundation of China. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work is supported by National Natural Science Foundation of China (No. 62306220).

References

[1] Jan Kocoń, Igor Cichecki, Oliwier Kaszyca, Mateusz Kochanek, Dominika Szydło, Joanna Baran, Julita Bielaniewicz, Marcin Gruza, Arkadiusz Janz, Kamil Kanclerz, Anna Kocoń, Bartłomiej Koptyra, Wiktoria Mieleszczenko-Kowszewicz, Piotr Miłkowski, Marcin Oleksy, Maciej Piasecki, Łukasz Radliński, Konrad Wojtasik, Stanisław Woźniak, Przemysław Kazienko, Chatgpt: Jack of all trades, master of none, Inf. Fusion 99 (2023) 101861.

[2] David C. Uthus, David W. Aha, Multiparticipant chat analysis: A survey, Artificial Intelligence 199 (jun.-ju) (2013) 106–121.

[3] Toan Nguyen-Mau, Anh-Cuong Le, Duc-Hong Pham, Van-Nam Huynh, An information fusion based approach to context-based fine-tuning of GPT models, Inf. Fusion 104 (2024) 102202.

[4] Rui Mao, Kai He, Xulang Zhang, Guanyi Chen, Jinjie Ni, Zonglin Yang, Erik Cambria, A survey on semantic processing techniques, Inf. Fusion 101 (2024) 101988.

[5] Zhengyuan Liu, Nancy Chen, Improving multi-party dialogue discourse parsing via domain integration, in: Proceedings of the 2nd Workshop on Computational Approaches to Discourse, Association for Computational Linguistics, Punta Cana, Dominican Republic and Online, 2021, pp. 122–127.

[6] Zhouxing Shi, Minlie Huang, A deep sequential model for discourse parsing on multi-party dialogues, in: Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI'19/IAAI'19/EAAI'19, AAAI Press, 2019.

[7] Jiaqi Li, Ming Liu, Min-Yen Kan, Zihao Zheng, Zekun Wang, Wenqiang Lei, Ting Liu, Bing Qin, Molweni: A challenge multiparty dialogues-based machine reading comprehension dataset with discourse structure, in: Donia Scott, Nuria Bel, Chengqing Zong (Eds.), Proceedings of the 28th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 2642–2652.

[8] Jimmy Wei, Kurt Shuster, Arthur Szlam, Jason Weston, Jack Urbanek, Mojtaba Komeili, Multi-party chat: Conversational agents in group settings with humans and models, 2023.

[9] Ling.Yu Zhu, Zhengkun Zhang, Jun Wang, Hongbin Wang, Haiying Wu, Zhenglu Yang, Multi-party empathetic dialogue generation: A new task for dialog systems, in: Smaranda Muresan, Preslav Nakov, Aline Villavicencio (Eds.), Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 298–307.

[10] Zhouxing Shi, Minlie Huang, A deep sequential model for discourse parsing on multi-party dialogues, in: Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI'19/IAAI'19/EAAI'19, AAAI Press, 2019.

[11] Ante Wang, Linfeng Song, Hui Jiang, Shaopeng Lai, Junfeng Yao, Min Zhang, Jinsong Su, A structure self-aware model for discourse parsing on multi-party dialogues, in: Zhi-Hua Zhou (Ed.), Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, International Joint Conferences on Artificial Intelligence Organization, Main Track, 2021, pp. 3943–3949.

[12] Ta-Chung Chi, Alexander Rudnicky, Structured dialogue discourse parsing, in: Oliver Lemon, Dilek Hakkani-Tur, Junyi Jessy Li, Arash Ashrafzadeh, Daniel Hernández Garcia, Malihe Alikhani, David Vandyke, Ondřej Dušek (Eds.), Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue, Association for Computational Linguistics, Edinburgh, UK, 2022, pp. 325–335.

[13] Qi Jia, Yizhu Liu, Siyu Ren, Kenny Zhu, Haifeng Tang, Multi-turn response selection using dialogue dependency relations, in: Bonnie Webber, Trevor Cohn, Yulan He, Yang Liu (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP, Association for Computational Linguistics, 2020, pp. 1911–1920, Online.

[14] Jiaao Chen, Diyi Yang, Structure-aware abstractive conversation summarization via discourse and action graphs, in: Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, Yichao Zhou (Eds.), Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics., Online, 2021, pp. 1380–1391.

[15] Alexander Chernyavskiy, Lidiia Ostyakova, Dmitry Ilvovsky, GroundHog: Dialogue generation using multi-grained linguistic input, in: Michael Strube, Chloe Braud, Christian Hardmeier, Junyi Jessy Li, Sharid Loaiciga, Amir Zeldes, Chuyuan Li (Eds.), Proceedings of the 5th Workshop on Computational Approaches to Discourse, CODI 2024, Association for Computational Linguistics, St. Julians, Malta, 2024, pp. 149–160.

[16] Angus Addlesee, Neeraj Cherakara, Nivan Nelson, Daniel Hernández García, Nancie Gunson, Weronika Sieińska, Marta Romeo, Christian Dondrup, Oliver Lemon, A multi-party conversational social robot using LLMS, in: Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction, HRI '24, Association for Computing Machinery, New York, NY, USA, 2024, pp. 1273–1275.

[17] Angus Addlesee, Neeraj Cherakara, Nivan Nelson, Daniel.Hernandez Garcia, Nancie Gunson, Weronika Sieińska, Christian Dondrup, Oliver Lemon, Multi-party multimodal conversations between patients, their companions and a social robot in a hospital memory clinic, in: Nikolaos Aletras, Orphee De Clercq (Eds.), Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, Association for Computational Linguistics, St. Julians, Malta, 2024, pp. 62–70.

[18] Jintao Wen, Dazhi Jiang, Geng Tu, Cheng Liu, Erik Cambria, Dynamic interactive multiview memory network for emotion recognition in conversation, Inf. Fusion 91 (2023) 123–133.

[19] Nuo Chen, Hongguang Li, Juhua Huang, Baoyuan Wang, Jia Li, Compress to impress: Unleashing the potential of compressive memory in real-world long-term conversations, 2024.

[20] Jie Liu, Yaguang Li, Shizhu He, Shun Wu, Kang Liu, Shenping Liu, Jiong Wang, Qing Zhang, Seq2set2seq: A two-stage disentangled method for reply keyword generation in social media, ACM Trans. Asian Low-Resour. Lang. Inf. Process 23 (3) (2024).

[21] Wen Zheng, Ke Zhou, Enhancing conversational dialogue models with grounded knowledge, in: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 709–718.

[22] Q Chen, W Wu, S Li, Exploring in-context learning for knowledge grounded dialog generation, in: Findings of the Association for Computational Linguistics: EMNLP 2023, Association for Computational Linguistics, 2023.

[23] Iulian V. Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, Joelle Pineau, Building end-to-end dialogue systems using generative hierarchical neural network models, in: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI'16, AAAI Press, 2016, pp. 3776–3783.

[24] Chen Xing, Yu Wu, Wei Wu, Yalou Huang, Ming Zhou, Hierarchical recurrent attention network for response generation, in: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI'18/IAAI'18/EAAI'18, AAAI Press, 2018.

[25] Weinan Zhang, Yiming Cui, Kaiyan Zhang, Yifa Wang, Qingfu Zhu, Lingzhi Li, Ting Liu, A static and dynamic attention framework for multi turn dialogue generation, ACM Trans. Inf. Syst. 41 (1) (2023).

[26] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, Language models are unsupervised multitask learners, 2019.

[27] Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, Bill Dolan, DIALOGPT : Large-scale generative pre-training for conversational response generation, in: Asli Celikyilmaz, Tsung-Hsien Wen (Eds.), Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Association for Computational Linguistics, 2020, pp. 270–278, Online.

[28] Jinpeng Li, Zekai Zhang, Xiuying Chen, Dongyan Zhao, Rui Yan, Stylized dialogue generation with feature-guided knowledge augmentation, in: The 2023 Conference on Empirical Methods in Natural Language Processing, 2023.

[29] Humza Naveed, Asad.Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Nick Barnes, Ajmal S. Mian, A comprehensive overview of large language models, 2023, ArXiv, abs/2307.06435.

[30] Hiroki Ouchi, Yuta Tsuboi, Addressee and response selection for multi-party conversation, in: Jian Su, Kevin Duh, Xavier Carreras (Eds.), Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Austin, Texas, 2016, pp. 2133–2143.

[31] Rui Zhang, Honglak Lee, Lazaros Polymenakos, Dragomir Radev, Addressee and response selection in multi-party conversations with speaker interaction rnns, in: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI'18/IAAI'18/EAAI'18, AAAI Press, 2018.

[32] Weishi Wang, Steven C.H. Hoi, Shafiq Joty, Response selection for multi-party conversations with dynamic topic tracking, in: Bonnie Webber, Trevor Cohn, Yulan He, Yang Liu (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP, Association for Computational Linguistics, 2020, pp. 6581–6591, Online.

[33] Cao Liu, Kang Liu, Shizhu He, Zaiqing Nie, Jun Zhao, Incorporating interlocutor-aware context into response generation on multi-party chatbots, in: Mohit Bansal, Aline Villavicencio (Eds.), Proceedings of the 23rd Conference on Computational Natural Language Learning, CoNLL, Association for Computational Linguistics, Hong Kong, China, 2019, pp. 718–727.

[34] Wenpeng Hu, Zhangming Chan, Bing Liu, Dongyan Zhao, Jinwen Ma, Rui Yan, Gsn: A graph-structured network for multi-party dialogues, in: International Joint Conference on Artificial Intelligence, 2019.

[35] Liang Qiu, Yizhou Zhao, Weiyan Shi, Yuan Liang, Feng Shi, Tao Yuan, Zhou Yu, Song-Chun Zhu, Structured attention for unsupervised dialogue structure induction, in: Yulan He Bonnie Webber, Yang Liu (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP, Association for Computational Linguistics, 2020, pp. 1889–1899, Online.

[36] Qi Song, Sheng Li, Ping Wei, Ge Luo, Xinpeng Zhang, Zhenxing Qian, Joint learning for addressee selection and response generation in multi-party conversation, in: ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, 2022, pp. 6587–6591.

[37] A Addlesee, N Cherakara, N Nelson, et al., A multi-party conversational social robot using LLMS, in: Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction, ACM, 2024, pp. 123–130.

[38] Weizhou Shen, Xiaojun Quan, Ke Yang, Generic dependency modeling for multi-party conversation, in: ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, 2023, pp. 1–5.

[39] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Christy Doran Jill Burstein, Thamar Solorio (Eds.), 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), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186.

[40] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, Yoshua Bengio, Graph attention networks, in: International Conference on Learning Representations, 2018.

[41] Yulei Yang, Dongsheng Li, Nenn: Incorporate node and edge features in graph neural networks, in: Sinno Jialin Pan, Masashi Sugiyama (Eds.), Proceedings of the 12th Asian Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 129, PMLR, 2020, pp. 593–608.

[42] Jinsong Su, Zhixing Tan, Deyi Xiong, Rongrong Ji, Xiaodong Shi, Yang Liu, Lattice-based recurrent neural network encoders for neural machine translation, in: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, AAAI'17, AAAI Press, 2017, pp. 3302–3308.

[43] David C. Uthus, David W. Aha, The ubuntu chat corpus for multiparticipant chat analysis, in: AAAI Spring Symposium: Analyzing Microtext, 2013.

[44] Khyati Mahajan, Sashank Santhanam, Samira Shaikh, Towards evaluation of multi-party dialogue systems, in: Thiago Ferreira Samira Shaikh, Amanda Stent (Eds.), Proceedings of the 15th International Conference on Natural Language Generation, Association for Computational Linguistics, Waterville, Maine, USA and virtual meeting, 2022, pp. 278–287.

[45] Lifeng Shang, Zhengdong Lu, Hang Li, Neural responding machine for short-text conversation, in: Chengqing Zong, Michael Strube (Eds.), Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Beijing, China, 2015, pp. 1577–1586.

[46] Chen Xing, Yu Wu, Wei Wu, Yalou Huang, Ming Zhou, Hierarchical recurrent attention network for response generation, in: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI'18/IAAI'18/EAAI'18, AAAI Press, 2018.

[47] Yawei Kong, Lu Zhang, Can Ma, Cong Cao, Hsan: A hierarchical self-attention network for multi-turn dialogue generation, in: ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, 2021, pp. 7433–7437.