MUDiC: A Dataset for Multi-User Dialogue and Collaboration in Chatbot Interaction

Nicolas Wagner*, **Cristina Luna-Jiménez**†, **Elisabeth André**†, **Wolfgang Minker** ‡, **Stefan Ultes***
*Natural Language Generation and Dialogue Systems Group, Otto-Friedrich University of Bamberg
{nicolas.wagner,stefan.ultes}@uni-bamberg.de
†Chair for Human-Centered Artificial Intelligence, University of Augsburg
{cristina.luna.jimenez,elisabeth.andre}@uni-a.de
‡Dialogue Systems Group, Ulm University
wolfgang.minker@uni-ulm.de

Abstract

We introduce MUDiC, a novel dataset on task-based multi-user interactions in chatbots. Unlike most traditional dialogue corpora that focus on one-to-one human–chatbot exchanges, this dataset captures conversations involving two human participants engaging with a dialogue system. The data include diverse conversational contexts such as shared group task, user intents, and mechanisms to deal with off-topic talk. MUDiC consists of 1,689 dialogue exchanges between 20 groups and the chatbot. Each session is annotated with user id, interaction turns, and intents and dialogue acts, enabling an analysis of group conversational dynamics. Consequently, the dataset aims to support tasks such as multi-user dialogue modelling, intent disambiguation, and moderation behaviour, which are relevant factors for the design of socially aware and fair chatbots. The dataset is available on Zenodo: https://doi.org/10.5281/zenodo.19037937.

Keywords: multi-user interaction, dialogue systems, group conversation datasets

1. Introduction

Chatbots have recently evolved from simple question–answering systems to advanced dialogue agents capable of engaging users in extended, context-maintaining discussions (Singh and Namin, 2025; McTear, 2022). Despite this progress, most existing research and datasets focus exclusively on dyadic interactions—that is, conversations between a single human user and a chatbot. In contrast, real-world communication often involves multi-user or group scenarios, involving multiple participants who work together on a shared task and need to coordinate their actions. This gap limits the capabilities of contemporary chatbots to operate effectively in settings where more than one user interacts with the same system. A major reason for the rather slow advancement is the absence of suitable training data (Kuhail et al., 2024). To address this limitation, we present MUDiC, a novel dataset designed to capture the dynamics of group interaction with a policy-operated chatbot. MUDiC contains conversations where two human users interact cooperatively to negotiate an appointment, discussing features such as location, date, and time. In summary, the main contributions of this work are:

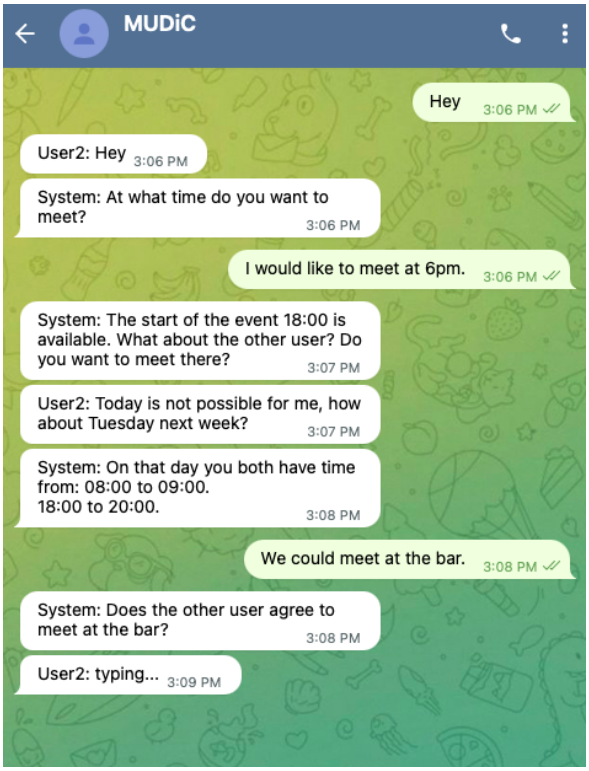

Figure 1: Visualisation of a conversation as experienced by User1.

1. Presentation and description of a dataset of authentic multi-user chatbot interactions in a collaborative task-oriented domain, consisting of 1,689 dialogue exchanges between users and the system.
2. Comprehensive annotations capturing key characteristics of the conversation domain, including extracted slot values from user input and logs of the decision-making of the chatbot.

Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026), pages 1925–1933
11–16 May 2026.

3. Analyses and metrics employing RoBERTa models for annotating different languages and evaluating transitions between conversation participants in multi-user environments.

Multi-user conversations pose unique challenges for modelling a dialogue policy. Unlike dyadic interactions, group conversations involve overlapping user requests and shifts in topics. The chatbot must be capable to track multiple user states, interpret ambiguous input, identify task-relevant information, and generate responses that are coherent and contextually appropriate to the group. Consequently, we build and release MUDiC to provide insights into designing chatbots that can assist during collaborative multi-user task-solving. An exemplary conversation is shown in Figure 1.

The remainder of the paper is structured as follows: Section 2 reviews the related work, summarising some of the potential existing challenges in multi-user chatbots and covering existing datasets. Section 3 outlines the methodology and the data acquisition protocol. Section 4 presents the different files containing the records and the validation procedures. Finally, Section 5 concludes the paper by highlighting the main takeaways and proposing directions for future research.

2. Related Work

Research in the field of dialogue systems and chatbots has mainly focused on one-to-one interactions, in which a user interacts directly with an agent (Rodríguez-Sánchez et al., 2025; Adamopoulou and Moussiades, 2020). However, in many daily life occasions, conversations happen in scenarios with multiple parties involved, such as in meetings, educational settings, or in other work-collaborative environments. Despite initial intuitions of applying the same approaches as in dyadic conversations, multi-user conversations involve new challenges that need to be addressed and studied. Previous work (Wagner et al., 2025; Ganesh et al., 2023) identified several challenges to be addressed when developing models to handle multi-user conversations:

- Participant roles: In a multi-user conversation, one of the first concerns is the existence of several interlocutors, making it important to adjust the dialogue flow and to identify the correct addressee(s). In this context, also in-group dynamics may be relevant for the user roles, with some users being more active versus some behaving more passively (Kraus et al., 2024).

- Turn-taking: Another key challenge in multi-user conversations is when the system is expected to take the turn. A turn-taking problem is typically associated to the context in which the chatbot is expected to engage in the conversation. For example, in some scenarios it could be intuitive to switch turn between user and system in a question-answer basis, such as in therapy; however, in teaching or educational settings, in which the chatbot is acting as a tutor, it is expected to interact more proactively or provide longer conceptual explanations than the learner, as it can be observed in recent corpus with dialogues generated in the educational context of a museum (Luna-Jiménez et al., 2024).

- Adaptation to context and user expectations: An additional challenge involves the continuos adaptation to the conversation, changing topics effectively and selecting appropriate responses and actions aligned the users goals, requests and preferences. As related work indicates that effective dialogue adaptation is improving the acceptance and efficiency of chatbots (Li et al., 2023), we aim to investigate different turn-taking mechanisms of the system.

Due to the increased interest in researching multi-user dialogue systems and to address beforementioned challenges, a number of datasets was published to study and analyse dynamics with the aim of developing systems that are able to handle group conversations. A literature review of the existing datasets in English is provided in (Mahajan and Shaikh, 2021), organising them by topic, size (number of dialogues, words and length in hours), modalities, tasks to study with the datasets (e.g., word sense disambiguation), level of formality and data collection or aggregation procedures. Additionally, the authors indicated a series of future directions to explore, such as dealing with different languages. As in many other research areas, advances in Large Language Models (LLMs) led to the development of performant multi-user chatbots that employ LLMs for dialogue modelling and management, as in (Mao et al., 2024). The training data used in that experiment, however, was synthetically generated by a user simulator, while our aim is to provide real user data. Despite the fact that LLMs produce grammatically correct text, there is still limited understanding of how genuine and natural the generated content truly is. While automatic metrics and human evaluations can measure aspects like fluency, coherence, and relevance, they may not always capture subtle nuances of authenticity or naturalness that mirror human language use.

Among the most prominent human-to-human dialogue datasets is MultiWOZ (Budzianowski et al., 2018). An extension of this corpus for multi-user interaction has been presented in (Jo et al., 2023).

Here, the authors used predicted rewrites of the utterances of the original corpus to simulate multi-user interaction. Although the rewritten utterances were designed to be semantically and pragmatically consistent with the original inputs, they may not fully reflect actual multi-user dialogue patterns found in real-world conversations. Our aim is thus to collect authentic data in a realistic scenario.

Messenger services are frequently used for data collection in dialogue research, as it offers a familiar and easily accessible platform for natural language interaction. While this method has been successfully applied to one-on-one conversations (Mamontov et al., 2025; Fiorentini et al., 2024), we consider it also appropriate for multi-user scenarios. The following section explains the methodology used to construct the dataset.

3. Building the Corpus

For the human-to-machine data collection, a multi-user chatbot was developed based on our previous work (Wagner et al., 2022). The system was embedded in a goal-oriented group negotiation scenario. More precisely, two users were given the task to agree on location, date, and time for a meeting. To enable assistance during task-solving, the chatbot was tracking user and group states, and generated responses based on four different dialogue policies. In this following, we will introduce the main steps and technical developments required to obtain the data.

3.1. Software and Recording Set-up

For recording the dialogues, Telegram was selected as the messaging application to create the multi-user chatbot communication. One advantage of this service is that it enables the setup and control of the chatbot via API. In our case, we implemented the chatbot's dialogue logic using the Rasa framework[1]. It features components for Natural Language Understanding and Dialogue Management. Figure 1 demonstrates a possible user experience, showing how the chatbot provides support during the interaction.

The dialogue policies of the chatbots were implemented using data-driven methods including the Transformer Embedding Dialogue (TED) model described in (Vlasov et al., 2019). In addition, a decision tree was employed as an alternative basis for decision-making in exceptional cases. Since this approach did not require specific hardware or software, we hosted the chatbot on a virtual machine running on an Apache2 server in a cloud-based setup. A more detailed description of the system can be found in (Wagner et al., 2022).

[1] https://rasa.com

3.2. Multi-User Chatbot Design

In this work, the multi-user chatbot relied on an intent-driven architecture. This means that it was necessary to define possible user intents as a first step. Defining possible user intents was essential to enable the chatbot to accurately interpret the goal or purpose behind each user's message and to select appropriate responses and actions.

Moreover, the response selection was guided by dialogue policies that determined the most appropriate action based on the context of the conversation. Our chatbot applied one of four available dialogue policies during each conversation. The policies determine the turn-taking behaviour of the chatbot and how assistance is offered to the group. Lastly, the system actions are described shortly.

3.2.1. User Intents Definition

In order to handle all potential interaction types that users may have with the system and each other during a group conversation, we defined the following users intents:

- Greetings and Endings: Some of the defined intents were specifically designed to initiate or conclude a conversation. In this category, the following intents are included:
  - 'start': signals the initiation of the conversation.
  - 'stop': indicates the end of the conversation.
  - 'greet': for greetings at the start of the conversation.
  - 'goodbye': when the conversation is considered finished.
- Affirmations or Negations: These intents represent input in which a user is either confirming or rejecting proposals made during the conversation. These intents play an important role in managing agreement and determining whether additional discussion is necessary to reach a final consensus. The following intents fall within this category:
  - 'affirm': for confirming a request (e.g., 'Yes').
  - 'deny': inverse of affirm, for rejecting offers or proposals.
- Event Agreement Intents: As one of the tasks in the multi-user scenario involves users agreeing on a meeting's specific date and location. To track how these interactions progressed, it was necessary to define intents focused on capturing relevant user inputs regarding scheduling. The following intents belong to this type:

  - 'date_suggestion': when the user suggests a possible date for the meeting.
  - 'earlier': to detect the intention of the user to meet earlier than the meeting time proposed.
  - 'later': to detect the intention of the user to meet later than the meeting time initially proposed.
  - 'location_planning': when the intention of the user is to propose and decide about places to meet.
  - 'move': to move the appointment of the decided event due to conflicts with other events or meetings in their calendars.

- Further Intents: Since the conversations were conducted in a real-world setting using the Telegram messenger to collect interactions for negotiating meetings, the system needed to consider and react to user intents outside the primary task. To address this, several intents were implemented to manage the following situations:
  - 'off_topic': when the utterance does not match to the group task, e.g. talking about the weather
  - 'out_of_scope': when the utterance does not match to any of the previously specified intents, e.g. when an emoji is used.
  - 'gif': when a GIF is sent as message by the users.
  - 'help': allowing the user to ask help from the administrator of the system, e.g, for troubleshooting technical problems.
  - 'otheruserreminder': for providing instructions about the experiment protocol, e.g., remind users to answer in the group-chat.
  - 'nlu_fallback': to handle errors in the Natural Language Understanding module, which indicates that a repetition or clarification by the user is necessary.
  - 'bot_challenge': triggered when a user asks the chatbot to perform a task unrelated to the scenario's main task.

3.2.2. Dialogue Policies

Four distinct strategies for dialogue management were developed and integrated into our chatbot to introduce variability in the chatbot's behaviour and to investigate how users discussed under these conditions. They model different interaction styles and influence the way the system moderates the negotiation process between group members. Moreover, we aimed to observe their impact on the dialogue flow and outcomes of the task.

- Baseline: The chatbot remains passive until a user explicitly requests help by typing messages such as 'Help'. It does not initiate any interaction.
- Notification: The system reacts whenever a user mentions a location, date, or time, providing status information on users' availability without specifying which group member is busy. It also periodically prompts users for date, time, or location if not yet provided.
- Private Negotiation: In cases of scheduling conflicts, the chatbot initiates a private message to the affected user asking if they would like to reschedule. If agreed, the calendar is adjusted and the group is informed. Also periodically prompts users for date, time, or location if not yet provided.
- Group Negotiation: A more direct strategy where the chatbot publicly notifies the entire group of conflicts by naming the user who is unavailable and asks if they would agree to reschedule, encouraging group discussion and resolution. Again, this policy includes periodical prompts for date, time, or location if not yet provided.

3.2.3. System Actions Definition

To enable the chatbot providing suitable responses according to the intents of the users in different moderation strategies, we also defined a set of actions that the system is able to perform. The system actions primarily manage user states and focus on conflict resolution during negotiations for scheduling meetings. They track which user is communicating, update confirmed information such as dates and locations after user agreement, and validate proposed dates against each users' calendar. Some actions handle denial of alternatives and reminders, offer possible dates, and determine whether system responses should be sent publicly or privately. These actions are designed to ensure cooperative negotiation, resolving scheduling conflicts by coordinating and confirming key meeting details among users.

3.3. Data Acquisition Protocol

Prior to data collection, each participant was informed about the data collection protocol and voluntarily signed a declaration of consent form, confirming their agreement to participate in the study and to have their data published. Additionally, participants were rewarded with €10 as compensation. They were instructed that the experiment was expected to take approximately one hour. The study was carried out in a between-subject design.

During the experiments, participants were assigned manually to a group chat in Telegram in teams of two, plus the chatbot to solve the given task. As already mentioned, the task consisted of negotiating an appointment. Users were asked to agreeing on a day, a time, and a meeting location within the next week. They were provided with a specially prepared Google calendar, which was already filled with a set of appointments, but also contained unoccupied time slots. Initially, the users were instructed how to use the system and that they could ask for assistance during their negotiation. The users were able to propose specific dates for a meeting or ask for suggestions. If a proposed date has been already blocked for at least one person, the chatbot informed about the conflict and also suggested a free time slot. If a user rejected this proposal, the system automatically marked the slot in the calendar. After having agreed on a time slot, users had to discuss the location of their meeting. For this, the bot is able to offer a list of options (e.g., gym or restaurant). If one or more users did not write anything for a longer time, the chatbot started to motivate them to rejoin the discussion. After finishing one negotiation, it was possible to repeat for another meeting. For the data collection, all teams randomly experienced two dialogue strategies during their experiment. Fallback actions were added to all policies in case of misunderstandings or if the user input could not be categorised. Since we manually added participants to the chat groups, we were able to ensure that all conditions had the same number of teams assigned and thus were balanced.

4. Dataset Format

The MUDiC dataset contains 1,689 authentic utterances from 38 group appointment discussions following the four dialogue strategies commented on before. The corpus has been recorded and curated to maintain a balanced distribution of the dialogue policies. This ensures fairness within the dataset and may help preventing bias in subsequent analyses or model training related to specific dialogue strategies or following them in generation tasks.

For the data collection, 40 individuals (45 % female) with an average age of 25.6 (SD = 8.5) were recruited, including students and external participants from diverse professional backgrounds. An overview of the descriptive statistics of MUDiC is shown in Table 1. The dataset uniquely identifies each dialogue using a 'dialogueID' and is organised in multiple files with the following content:

- rawData.csv: This file contains the raw data of the conversations after anonymisation. Specifically, it has seven columns: 'id', 'sender_id', 'type_name', 'timestamp', 'intent_name', 'action_name', and 'data'. Each of the columns represents:
  - 'id': this field stands for the identifier of the state of the conversation, including any intent or system actions (even for those passive, e.g. 'action_listen').
  - 'sender_id': this field is the internal Telegram ID assigned to the participants for the application, so there is one per dialogue or chat.
  - 'type_name': this field contains the name of the events that happen during the dialogue and the interactions.
  - 'timestamp': this field has the time at which each of the events of the dialogue occurs.
  - 'intent_name': this column contains the intent of the user, hence only available when there is an event triggered by the user, being null otherwise.
  - 'action_name': This field has information about the actions of the system, which could include filling in an 'slot' or taking an action by the system.
  - 'data': this field contains the history of the conversation and the fulfilled slots and events. One example of this field is: '{'event': 'slot', 'timestamp': 1625914004.5979762, 'name': 'confirmed_location', 'value': 'gym'}'.
  - 'dialogueID': this field indicates the dialogue identifier. The format of the identifiers is '$< NumberOfDialogue > \_Session\_ < PolicyID >$', in which $PolicyID$ is 1 for the 'Baseline', 2 for the 'Notification', 3 for the 'Private Negotiation', and 4 for the 'Group Negotiation'.

Description	Amount
Number of dialogues	37
Number of utterances	1689
Users interactions	919
System interactions	690
Admin interactions	80
Vocabulary Size	668
Description	**Count**
Avg./Std. Words per User Utterance	3.43 ± 3.51
Avg./Std. Words per System Utterance	17.82 ± 16.63
Avg. Turns of Users	12.00
Avg. Turns of System	19.56

Table 1: Descriptive statistics of conversations in the MUDiC dataset.

- databaseComplete.csv: This file contains all the dialogues recorded on the different sessions. It has 8 columns with the 'DB_id', 'dialogue_ID', 'utterance_id', 'timestamp', 'utterance_text', 'dialogue_intent', 'system_action' and 'party'. *Both 'DB_id'* and *'dialogue_id'* are identifiers of the database and session of the recordings; next, the *'utterance_id'* column contains the index of each utterance in the database. Next, the *timestamp* column contains the timestamp informing when the utterance took place. The column named *'utterance_text'* contains the actual text typed by the user or the system. The following column is reserved for the *'user_intent'* which are described in detail in Section 3.2.1. The *'system_action'* column contains the actions that the system could perform according to the selected dialogue policy. Lastly, the *'party'* indicates which role the speaker had in the interaction.

- languageMetrics.csv: This file contains the results of the metrics performed as part of the Language Detection of the technical validation. It contains information of the utterance, plus two additional columns with the language of the utterance and the confidence score. The subsequent analysis of collected data is described in the next section.

4.1. Participant Transition Analysis

To better understand the overall interaction dynamics of the collected conversations, we computed an average transition matrix combining the results from the four dialogue policies. The aim was to summarise the probability distribution of utterance transitions between participants. Higher values indicate more frequent turn-taking patterns between these party members, which is illustrated in Table 2. The averaged matrix reflects the general distribution of utterance transitions between different participants ($User_1$, $User_2$, $System$, and $Admin$) throughout the dialogues. The individual matrices for each dialogue policy will be listed in the appendix. In summary, the selected policy had influence on which addressee the participants selected.

Moreover, Figure 2 displays the Sankey diagram of the dialogue flow in which we consider as source the $utterance_t(U_t)$ and as target the subsequent $utterance_{t+1}(U_{t+1})$. Each of the lines represents the following response by one of the users of the system ($User_1$ or $User_2$), the agent ($System$), or the administrator of the system ($Admin$). The administrator interventions are limited to those cases where users needed off-task assistance, instructions, or to reset the conversation. However, they are included in the graph for a better understanding

$U_t \backslash U_{t+1}$	$Start$	$User_1$	$User_2$	$System$	$Admin$
$Start$	–	78.61	–	–	21.39
$User_1$	–	13.95	17.44	67.25	1.37
$User_2$	–	15.36	17.95	64.02	2.19
$System$	–	40.29	44.97	9.53	5.72
$Admin$	–	30.85	16.24	32.06	21.85

Table 2: Average matrix of users' utterance addressees considering $utterance_t$ and $utterance_{t+1}$ in percentage across the four dialogue policies.

of the dialogue flow in each scenario and to analyse the interactions between participants and the system. Additionally, we included an initial state called $start$ as the starting point of the dialogue at timestep 0. The values (and width of the interconnection lines) are normalised by the total number of interactions starting from each party (including Admin and the System) at the $utterance_t$. Again, we display an average value for all dialogue policies.

4.2. Post-Processing

In addition to the acquisition of the data, several filters were applied in order to anonymise the data and identify in which language the participants spoke, as well as to detect informal expressions.

Anonymisation : As a first step, we ensure that the data was anonymised by inspecting when the name of one of the participants appeared in the dialogues. Since one of the features of the chatbot was to address the participants by their first names, this occurred frequently. To remove any names, we lowercase the text in the utterances and checked whether the names appeared in our vocabulary. In case the name appeared in the vocabulary, it was substituted by the token $< FEMALE\_NAME >$ when the name was a female name and by the token $< MALE\_NAME >$ when it was a male name.

Text Normalisation. For normalising the text, we only applied conversion to lowercase given that we were interested in maintaining the language as natural as possible, and also considering that the emojis or other ways of communication would be relevant for the dialogue. In particular, no changes on punctuation was made.

Language Detection. As the participants of the experiments were English and German speakers, we performed an analysis of the languages employed. In a first filtering round, we employed a XLM-RoBERTa transformer model with a final classification head on top to detect up to 20 different languages (Conneau et al., 2020)[2] to extract a first estimation of the existing languages in the dataset. We obtained the output that appears in Table 3. Af-

[2]'papluca/xlm-roberta-base-language-detection' model available on Huggingface

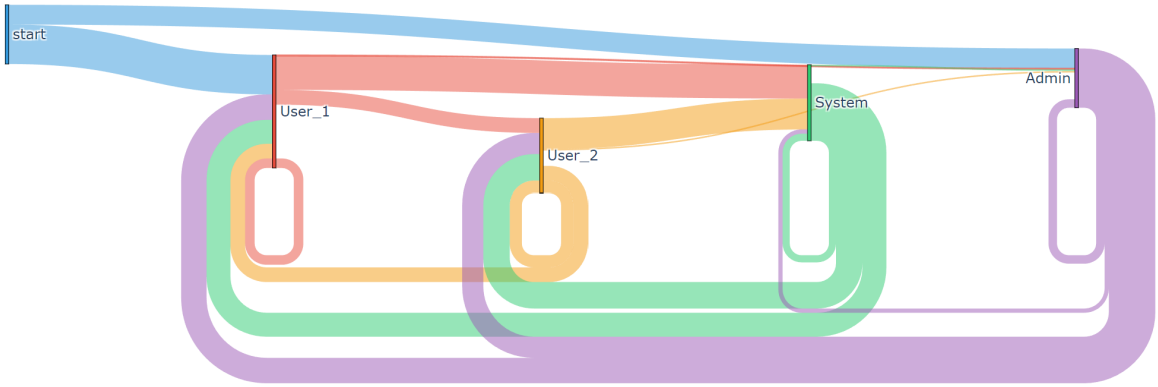

Figure 2: Averaged Sankey diagram for the dialogue policies considering connections of $utterance_t \rightarrow utterance_{t+1}$

ter this first analysis in which we saved the detected language and the confidence of the model, we performed a second manual filtering round in which we inspected the utterances with a number of repetitions lower than 16, plus those non-expected languages as Urdu and Swahili. In this second round, we substituted the wrong predictions and added 1.0 in the score column for those values manually annotated and accepted. At the end of the scanning, we finished with only two languages: English and German, with 1659 and 30 utterances, respectively. In the second manual inspection, we observed that several of the misclassified utterances contained abbreviations (e.g., 'thx', 'Wbu'), laughs (e.g., 'hahaha'), numbers (e.g. '8'), natural informal language (e.g., 'yoo'), or emojis, which may explain why other, unrelated languages were predicted in a few cases.

5. Conclusion and Future Work

In this paper, we presented the multi-user dataset MUDiC in the task-based domain of appointment negotiation. The data was acquired through interactions of two users with a chatbot across four different dialogue policies. The corpora contains informal utterances and was analysed to assess temporal evolution between dialogue exchanges. The analysis revealed a strong predominance of user–system interactions, with individual user utterances commonly followed by system responses. This pattern indicates that the conversation flow is largely reactive, centring on the system's role as a mediator or assistant rather than as an initiator in multi-user exchanges. Although administrative and cross-user transitions occur less often, their presence demonstrates that some degree of coordination needs to be maintained. In conclusion, MUDiC provides a valuable linguistic resource to explore human-computer interactions and changes in user behaviour with respect to the system policy. For future work, we intend to conduct experiments involving three or more users, and potentially more languages and conversation domains in the interactions. We also plan to investigate the integration of policy-driven dialogue control with LLMs, aiming to combine structured dialogue management with the generative capabilities of such models.

ISO code	Language	Number of Samples
en	English	1409
ur	Urdu	132
sw	Swahili	93
de	German	27
hi	Hindi	15
it	Italian	5
nl	Dutch	4
bg	Bulgarian	1
es	Spanish	1
pt	Portuguese	1
pl	Polish	1

Table 3: Classification of utterances per language.

6. Limitations

Although the dataset contains realistic and natural conversations of users, it also has certain limitations. First, although the dataset was collected in English, the test users were mainly native German speakers. Moreover, users were asked to interact within a controlled, predefined task scenario. Additionally, coordinating the data collection process proved challenging, as organising suitable group sessions was difficult and ultimately resulted in a rather small dataset size.

7. Acknowledgments

This contribution receives funding by the FORSocialRobots project (BFS, Bavarian Research Foundation, grant number AZ1594-23).

8. Bibliographical References

Eleni Adamopoulou and Lefteris Moussiades. 2020. An overview of chatbot technology. In *IFIP international conference on artificial intelligence applications and innovations*, pages 373–383. Springer.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Ilaria Fiorentini, Marco Forlano, and Nicholas Nese. 2024. Towards the WhAP corpus: A resource for the study of Italian on WhatsApp. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 16659–16663, Torino, Italia. ELRA and ICCL.

Ananya Ganesh, Martha Palmer, and Katharina Kann. 2023. A survey of challenges and methods in the computational modeling of multi-party dialog. In *Proceedings of the 5th Workshop on NLP for Conversational AI (NLP4ConvAI 2023)*, pages 140–154, Toronto, Canada. Association for Computational Linguistics.

Yohan Jo, Xinyan Zhao, Arijit Biswas, Nikoletta Basiou, Vincent Auvray, Nikolaos Malandrakis, Angeliki Metallinou, and Alexandros Potamianos. 2023. Multi-user MultiWOZ: Task-oriented dialogues among multiple users. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 3237–3269, Singapore. Association for Computational Linguistics.

Matthias Kraus, Stina Klein, Nicolas Wagner, Wolfgang Minker, and Elisabeth André. 2024. A pilot study on multi-party conversation strategies for group recommendations. In *Proceedings of the 6th ACM Conference on Conversational User Interfaces*, CUI '24, New York, NY, USA. Association for Computing Machinery.

Mohammad Amin Kuhail, Imran Taj, Saifeddin Alimamy, and Bayan Abu Shawar. 2024. A review on polyadic chatbots: trends, challenges, and future research directions. *Knowl. Inf. Syst.*, 67(1):109–165.

Han Li, Renwen Zhang, Yi-Chieh Lee, Robert E Kraut, and David C Mohr. 2023. Systematic review and meta-analysis of ai-based conversational agents for promoting mental health and well-being. *NPJ Digital Medicine*, 6(1):236.

Cristina Luna-Jiménez, Manuel Gil-Martín, Luis Fernando D'Haro, Fernando Fernández-Martínez, and Rubén San-Segundo. 2024. Evaluating emotional and subjective responses in synthetic art-related dialogues: A multi-stage framework with large language models. *Expert Systems with Applications*, 255:124524.

Khyati Mahajan and Samira Shaikh. 2021. On the need for thoughtful data collection for multi-party dialogue: A survey of available corpora and collection methods. In *Proceedings of the 22nd annual meeting of the special interest group on discourse and dialogue*, pages 338–352.

Danila Mamontov, Alexey Karpov, and Wolfgang Minker. 2025. A multilingual telegram chatbot for mental health data collection. In *Proceedings of the 27th International Conference on Multimodal Interaction*, pages 794–796.

Manqing Mao, Paishun Ting, Yijian Xiang, Mingyang Xu, Julia Chen, and Jianzhe Lin. 2024. Multi-user chat assistant (muca): a framework using llms to facilitate group conversations.

Michael McTear. 2022. *Conversational ai: Dialogue systems, conversational agents, and chatbots*. Springer Nature.

María Jesús Rodríguez-Sánchez, Zoraida Callejas, Angel Ruiz-Zafra, and Kawtar Benghazi. 2025. Breaking the limits of chatbot development: Api-driven multi-domain chatbot generation empowered by generative ai. *Computing*, 107(11):1–32.

Sonali Uttam Singh and Akbar Siami Namin. 2025. A survey on chatbots and large language models: Testing and evaluation techniques. *Natural Language Processing Journal*, 10:100128.

Vladimir Vlasov, Johannes E. M. Mosig, and Alan Nichol. 2019. Dialogue transformers. *CoRR*, abs/1910.00486.

Nicolas Wagner, Matthias Kraus, Wolfgang Minker, David Griol, and Zoraida Callejas. 2025. A survey on multi-user conversational interfaces. *Applied Sciences*, 15(13):7267.

Nicolas Wagner, Matthias Kraus, Tibor Tonn, and Wolfgang Minker. 2022. Comparing moderation strategies in group chats with multi-user chatbots. In *Proceedings of the 4th Conference on Conversational User Interfaces*, CUI '22, New York, NY, USA. Association for Computing Machinery.