Do LLMs Understand Dialogues? A Case Study on Dialogue Acts

Ayesha Qamar, Jonathan Tong and **Ruihong Huang**
Department of Computer Science and Engineering
Texas A&M University, College Station, TX
{ayesha, tongjo, huangrh}@tamu.edu

Abstract

Recent advancements in NLP, largely driven by Large Language Models (LLMs), have significantly improved performance on an array of tasks. However, Dialogue Act (DA) classification remains challenging, particularly in the fine-grained 50-class, multiparty setting. This paper investigates the root causes of LLMs' poor performance in DA classification through a linguistically motivated analysis. We identify three key pre-tasks essential for accurate DA prediction: **Turn Management**, **Communicative Function Identification**, and **Dialogue Structure Prediction**. Our experiments reveal that LLMs struggle with these fundamental tasks, often failing to outperform simple rule-based baselines. Additionally, we establish a strong empirical correlation between errors in these pre-tasks and DA classification failures. A human study further highlights the significant gap between LLM and human-level dialogue understanding. These findings indicate that LLMs' shortcomings in dialogue comprehension hinder their ability to accurately predict DAs, highlighting the need for improved dialogue-aware training approaches.

1 Introduction

A Dialogue Act (DA) represents an utterance's communicative function (Searle, 1969). Some common examples are question and request. Understanding DAs is a fundamental first step in analyzing and comprehending dialogues (Stolcke et al., 2000). Understanding the role an utterance plays in the broader context of a dialogue requires not just the understanding of the semantic content of the utterance but also how it relates to the previous utterances, the speaker interactions, and what effect it was supposed to have on the addressee. Two utterances with the same semantic content could convey different meanings based on the context and the speaker's role. For example, floor mechanisms can easily be confused with short response utterances.

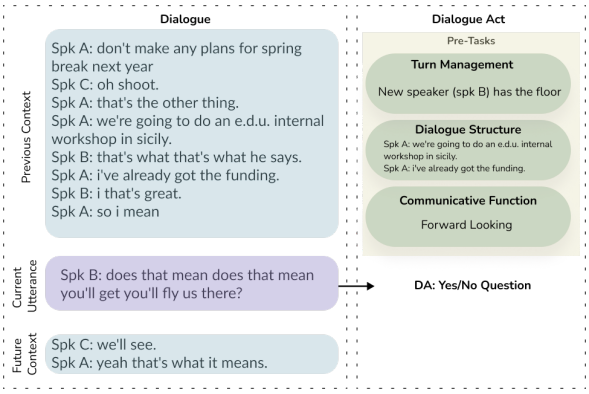

Figure 1: To correctly predict that the current utterance is asking a *Yes/No Question*, identifying the speaker roles, relevant context selection, and understanding the high-level communicative function is needed.

Large Language Models (LLMs) have revolutionized the field of artificial intelligence. They have shown remarkable performance on many unseen tasks in a zero-shot setting (Kojima et al., 2022), primarily due to their vast number of parameters, which allow them to store substantial amounts of information (Roberts et al., 2020), as well as the extensive datasets on which they are pre-trained. This capability has led to their adoption in numerous applications, particularly within the dialogue domain. Some applications include conversational assistants, chatbots, dialogue summarization, and dialogue-state tracking systems. A common task shared by all these applications is understanding the speaker's intentions.

Even with LLMs shown to do well on many unseen tasks (Wang et al., 2023), it remains a question if LLMs perform well on fine-grained DA classification. Through a comprehensive study across two benchmark datasets, we show that there is a huge performance gap between smaller fine-tuned models on DA classification and large models in zero and few-shot in-context learning setting. Additionally, we explore the role of multimodal audio

features, revealing that while they offer improvements, LLMs still fail to fully leverage prosodic cues.

Looking at dialogue acts from a linguistic lens, we introduce three fundamental *pre-tasks* necessary for accurate DA classification. The first is *Turn Management*, which helps identify speaker turn-taking roles in dialogues. The Second is *Communicative Function Identification*, which gives the high-level function an utterance plays. There are two main types: backward-looking, when an utterance refers to past context, and forward-looking, when it shapes future discourse. Lastly, the *Dialogue Structure* aims to capture direct speaker interactions such as question-answer pairs. We evaluate models on these tasks by repurposing existing annotations to create three pre-task datasets. Our findings reveal a strong empirical correlation suggesting that errors in these pre-tasks are associated with errors in DA classification.

Additionally, LLMs struggle with these fundamental tasks and perform comparatively or worse than naive non-parametric baselines on these pre-tasks–highlighting their lack of dialogue comprehension abilities. Particularly, our analysis showed that LLMs (1) struggle to identify speaker turn-taking roles, (2) mostly rely on nearer utterances for context and consequently fail to capture long dependency relations between utterances, and (3) are biased to view utterances as serving a backward-looking function even when they are not. We also conducted a human study that showed humans do not find these pre-tasks difficult but LLMs lag significantly behind human-level performance. Our findings highlight the need for better dialogue-aware training strategies to bridge the gap between human and machine dialogue understanding.

2 Related Work

DA Classification Prior work has used hierarchical architectures to encode the surrounding utterancs (Liu et al., 2017; Kumar et al., 2018a). (Raheja and Tetreault, 2019) use two utterance and conversation-level RNNs with contextual attention to generate utterance representations. Kumar et al. (2018a) treat DA classification as a sequence labeling task and incorporate a CRF layer to aid in learning class associations. Incorporating a wider context is particularly beneficial for underrepresented classes (Żelasko et al., 2021; Ahmadvand et al., 2019). Injecting speaker information has also been shown to improve DA performance. He et al. (2021) enrich utterance representations by learning speaker turn embeddings. Qamar et al. (2023) propose a graph neural network with utterance and speaker nodes to capture speaker interactions. While Shang et al. (2020) modify the final CRF layer to account for speaker changes. Audio features also play an important role in disambiguating certain DA labels as demonstrated by performance gains achieved using multimodal models that incorporate audio features (Miah et al., 2023).

LLMs for Dialogue Understanding LLMs have been applied for many broader dialogue-related tasks. These include applications for task-oriented dialogues such as dialogue state tracking (Luo et al., 2024; Pan et al., 2023; Heck et al., 2023) and intent detection (Arora et al., 2024). Another application of LLMs is in dialogue summarization. Laskar et al. (2023) study LLMs' capabilities on long meeting summarization by truncating the original dialogue into chunks and using different techniques to combine the results. Understanding emotions is essential for dialogue systems to create a positive user experience (Liu et al., 2021). Zhang et al. (2024) use chain of thought reasoning with explicit emotion identification for emotion-sensitive and empathetic response generation. Kang et al. (2024) show that LLMs exhibit certain biases and on their own are inadequate for emotional support conversations. Moreover, LLMs struggle with nuances of conversations such as understanding emphasized sentences (Lin and Lee, 2024). While prior work has studied various aspects of dialogue understanding for LLMs, there has been no extensive study of their performance on DA recognition in the challenging setting of multi-party dialogues.

3 Experimental Details

3.1 Datasets

We experiment with two commonly studied corpora for DA classification in natural dialogues–The Switchboard Dialog Act corpus (SwDA) and ICSI Meeting Recorder Dialog Act (MRDA). SwDA (Jurafsky, 1997) is a two-party dialogue dataset where participants were asked to converse on a pre-specified topic. MRDA (Shriberg et al., 2004)

Dataset	|D|	|U|
MRDA	11	15k
SwDA	19	4.5k

Table 1: |D|, |U| give the number of dialogues and utterances in the test set respectively.

is a multiparty dataset consisting of 75 naturally occurring meetings, where each meeting is around an hour long. Both SwDA and MRDA have been annotated for fine-grained DA classes. SwDA follows a scheme of 43 classes while MRDA has 50 [1]. Both datasets contain corrected transcripts along with audio recordings.

Historically, the coarse-grained DA labels have received much attention (Raheja and Tetreault, 2019; He et al., 2021) with performance on the 5-class label set surpassing 90% accuracy. Recently, analysis and experiments have been done on the much more challenging task of fine-grained DA classification (Żelasko et al., 2021; Qamar et al., 2023; Miah et al., 2023). Since good performance on the fine-grained DA classes is needed for a deeper dialogue understanding, we perform our analysis under this setting.

3.2 Models

We compare LLMs with several smaller SFT models. *RoBERTa$_{base}$* (Liu et al., 2019) is a simple roberta model with a linear layer on top, *BiLSTM+CRF* is a BiLSTM model with CRF (Kumar et al., 2018b) and *BiLSTM SelfAtt+CRF* also includes self attention (Raheja and Tetreault, 2019), *Turn Modeling* learns two speaker embeddings on top of RoBERTa model (He et al., 2021), and finally the *Speaker Graph* model learns speaker interactions through a graph neural network (Qamar et al., 2023). All of these models use a RoBERTa backbone and have under 160 million parameters. For LLMs, we experiment with various open-source and proprietary models, covering a wide range of model sizes.

3.3 Prompting LLMs

All the experiments in this paper are conducted by framing the task as a classification problem. The speaker names are prepended to the utterances to make the input speaker aware. Specific prompts used are provided in the appendix. The following prompt template is used in all cases:

Instruction:
[Task Description]
[Definitions of the Class Labels]
[In-context Examples]

Input:
[Previous Context Utterances]
[Current Speaker: Current Utterance]
[Future Context Utterances]

3.4 Audio Features

Intonation plays an important part in spoken language as it conveys meaning (Brazil, 1997). For instance, *Backchannel*, *Acknowledgment*, and *Accept* often look similar in their textual representation but have distinct audio characteristics, making them difficult to disambiguate using text alone (Shriberg et al., 2004). Consequently, text-only models frequently fail to make these distinctions. To address this limitation, we conducted experiments where audio was provided alongside the dialogue transcript, aiming to determine whether the LLM could leverage differences in pitch to infer the underlying speaker intention.

3.5 Implementation Details

MRDA and SwDA use anonymized speaker IDs to distinguish speakers. Following Kim and Vossen (2021), we also assign names to the speaker IDs. We use the top US names used in the last century for this purpose [2]. The exact prompts and model cards used for all the tasks are given in the Appendix F. We include in-context examples in all the prompts: 4 examples for DA, 3 for Turn Management, 5 for Communicative Function, and 4 for Dialogue Structure. The examples are the maximum number of examples we can add before we see a performance drop on the validation set or we run into GPU memory issues. All the experimental results reported are an average of 3 random seeds except the GPT models, since those are not open-source and incur substantial financial costs to run for multiple seeds. The experiments with all the open-source models were run using the LLaMA-Factory library [3] (Zheng et al., 2024) on 2 A100 GPUs.

4 LLM performance on DA classification

This section analyzes the results of various-sized LLMs compared to fully supervised, fine-tuned

[1] We follow the fine-grained classes as presented in (Qamar et al., 2023)

[2] https://www.ssa.gov/oact/babynames/decades/century.html

[3] https://github.com/hiyouga/LLaMA-Factory

	Model	MRDA				SwDA			
		Precision	Recall	F1	Accuracy	Precision	Recall	F1	Accuracy
Fine-tuned	RoBERTa$_{base}$	37.14	39.58	36.33	64.24	47.12	51.24	46.65	72.22
	BiLSTM+CRF	36.09	32.87	32.69	65.38	59.11	53.69	54.91	79.10
	BiLSTM SelfAtt+CRF	34.32	32.19	31.21	63.66	54.41	51.14	51.07	74.44
	Turn Modeling	43.52	38.92	38.77	67.0	62.48	56.9	57.96	81.0
	Turn Aware Speaker Graph	44.53	39.11	40.06	66.32	63.72	57.36	58.81	80.86
Few-shot	Gemma-7B	13.68	8.0	8.27	28.34	11.13	15.90	7.77	21.18
	Mistral-7B	6.30	6.05	4.65	13.44	13.64	13.64	10.29	20.2
	LLaMA-3.1-8B	18.47	13.02	12.09	20.54	16.45	16.71	11.38	21.36
	LLaMA-3.1-70B	26.49	22.36	19.17	31.26	21.39	26.53	19.59	39.96
	GPT-3.5	22.75	20.86	18.29	31.46	15.83	22.17	14.61	31.63

Table 2: Fine-grained DA performance under fine-tuned and zero-shot settings on both the MRDA and SwDA datasets using only text input. Fine-tuned results taken from Qamar et al. (2023). For the LLMs, we have used the instruction-tuned version of the models.

Model	Precision	Recall	F1	Accuracy
GPT-4	26.58	31.84	24.73	31.40
GPT-4 Audio	28.30	35.53	28.27	41.83

Table 3: GPT-4 performance on MRDA with and without audio. For the audio model, we also provide the transcript.

smaller models. Not only can LLMs not beat smaller models, but they also perform significantly worse across both datasets. As shown in Table 2, LLaMA-70B, with 70 billion parameters, fails to outperform a simple RoBERTa-base baseline model, which has approximately 125 million parameters[4]—making it about 560 times smaller in size.

Performance evaluation on MRDA more faithfully represents how well models can do in real-world conversations because MRDA meetings are unstaged meetings with multiple speakers taking part in the conversation. Therefore, we analyze LLM performance on MRDA in extensive detail. Detailed numbers can be seen in the confusion matrices in Appendix B.

4.1 Error Patterns

Understanding speaker intent heavily depends on utilizing the context provided by prior utterances. For instance, *Floor Grabber* and *Hold* share a similar vocabulary but serve distinct functions: the former refers to utterances where a speaker not currently holding the floor attempts to gain it, while the latter marks instances where the speaker is explicitly granted the floor. Without considering preceding utterances and the speaker roles, distinguishing between these two classes is nearly impossible. The frequent misclassification of these labels by LLMs highlights their limited ability to comprehend complex dialogue dynamics.

Furthermore, LLMs often struggle to look beyond the surface or syntactical structure of utterances, limiting their ability to grasp deeper meanings. They frequently confuse *Rhetorical Question* with *Wh-Question*[5]. While the former does not elicit a response, the latter does. Similarly, *Accept* and *Acknowledgment*—short positive responses—are often mistaken for *Backchannels*. Since dialogues are a collaborative task requiring alternating speakers, LLMs should accurately interpret these interactions for effective DA classification. Although incorporating the audio modality led to improved performance as shown in Table 3, the model still struggled to reliably distinguish between the aforementioned cases, illustrating its limited ability to fully leverage audio information.

5 Theoretical Perspective on Dialogue Acts

Most DA frameworks identify *dimensions* of an utterance that play a part in determining its DA label. While these dimensions can vary depending on the use case and the underlying dataset, several key aspects remain consistent across many frameworks. In this section, we outline the most common overlapping characteristics of a dialogue that are crucial in determining the DA of a given utterance. We call these *pre-tasks*.

[4] https://huggingface.co/transformers/v2.4.0/pretrained_models.html

[5] Questions that seek specific information and typically include "wh" words such as what, why, which, or who.

Communicative Function The DAMSL framework, adapted with modifications for annotating the SwDA and later the MRDA corpus, classifies DAs into three levels: Utterance Features, Forward and Backward Looking Communicative Functions (Core and Allen, 1997). Utterance Features pertain to the content and form of an utterance. These are further divided into two subcategories: Information Level, which focuses on whether the utterance addresses task-related content or communication management, and Communicative Status, which records whether the utterance is intelligible and successfully completed (Allen and Core, 1997). The Communicative Function describes how an utterance connects to the prior discourse as a Backward Looking Function (BLF) or how it impacts the future beliefs and actions of participants, thereby shaping the discourse, as a Forward Looking Function (FLF). To classify a response into finer labels such as Accept, Partial Accept, or Affirmative Answer, it is first necessary to recognize that the utterance performs a Backward Looking Function. Similarly, when a speaker poses a question, they influence the addressee's future actions[6]. Thus, recognizing that the utterance serves a Forward Looking Function is important to correctly classify it into a specific question label.

Turn Taking Turn management refers to the allocation of speaker roles within a dialogue (Sacks et al., 1974). It is a fundamental aspect of conversations, enabling participants to take turns speaking and ensuring the progression of the conversation. Bunt et al. (2012) recognize Turn Management as one of the dimensions used to determine a DA, while Bunt (1994) classify turn assignment as part of the broader category of social context. Many DA tags rely on an understanding of when participants hold the floor, are attempting to gain it, or wish to express understanding through *Backchannels* without seeking to take the floor. In unscripted conversations involving imperfect agents, overlapping turns are common (Bel-Enguix and Jiménez-López, 2006), further complicating the dynamics of dialogue. The intonation and syntactical features of an utterance determine when the current turn is expected to end (Holtgraves, 2013). Furthermore, adding turn awareness into models has been shown to improve DA performance (He et al., 2021). These complexities emphasize the importance of designing systems for DA prediction that can accurately interpret turn management within a conversation.

Dialogue Structure Various DA annotation frameworks include some form of tracking the dialogue structure. (Popescu-Belis, 2005) argue that Adjacency Pair[7] (AP) captures important aspects of an utterance function that cannot be solely inferred through speech acts. Bunt et al. (2012) refer to the strong coupling of a DA with its preceding DAs in the form of *Dependence Relations*. In particular, the *functional dependence relation* is defined as *'relation between a given dialogue act and a preceding dialogue act on which the semantic content of the given dialogue act depends due to its communicative function.'* Context is of paramount importance for DA disambiguation and has been shown to improve DA performance (Żelasko et al., 2021). Boyer et al. (2009) showed that adjacency pairs can be used to distill implicit dialogue structure. Moreover, incorporating explicit dialogue structure has been shown to improve DA classification performance (Xu et al., 2022; Shi and Huang, 2019).

In summary, for a system to perform accurate DA classification, it must be capable of understanding the communicative function of an utterance, recognizing how turn-taking unfolds in natural dialogues, and extracting the necessary structural information from prior context.

6 Dialogue Understanding of LLMs

In this section, we systematically evaluate LLM performance on the three pre-tasks to gain insight into the sources of their poor performance. For discussion, we report the performance of the LLaMA-70B model for all these tasks while additional model performance can be found in Appendix C.

6.1 Communicative Function of an Utterance

The DAMSL framework assigns DAs into one of the two communicative functions along with other dimensions that are largely concerned with the form of the utterance. We are interested in the communicative function an utterance holds, therefore the other dimensions have been merged into a single *'Other'* dimension. To create a *Commu-*

[6]The addressee may choose to respond to the question or not.

[7]An Adjacency Pair is a set of utterances grouped into first and second parts that are spoken by different speakers. Common examples include greeting-greeting and offer-accept pairs.

Communicative Function (CF)			Turn Management (TM)		
Backward-Looking	Forward-Looking	Other	New Floor	Floor Continuation	No Floor
6232	6882	1950	5796	7047	2221

Table 4: The number of utterances belonging to each class in the MRDA test set for the *Communicative Function* and *Turn Management* pre-tasks.

Label	LLM			Naive Baseline		
	Precision	Recall	F1	Precision	Recall	F1
Backward-Looking	53.61	74.45	62.34	93.85	0.5	65.24
Forward-Looking	76.86	44.97	56.74	63.10	99.85	77.33
Other	44.46	54.31	48.89	96.36	42.15	58.65
Macro Avg	58.31	57.91	55.99	**84.44**	**64.0**	**71.76**
Accuracy		58.38			**67.08**	

Table 5: LLaMA-70B model performance on identifying the communicative function of an utterance. A rule-based baseline can perform better than the LLM.

Algorithm 1 Communicative Function from DA

Input: $DA_{general}, DA_{specific}$
Parameter: $f_{Dimension}()$, a function that returns the dimension a given DA belongs to.
Output: Communicative Function

if $DA_{general}$ == Question **then**
  $function$ = 'Forward Communicative Function'.
else
  $function = f_{Dimension}(DA_{specific})$
end if
if $function \in$ {'Information Level', 'Communicative Status'} **then**
  $function$ = 'Other' *//merge them into one category*
end if
return function

nicative Function (CF) task, we use the hierarchy presented in SWBD-DAMSL with manually allocating a function to any new tags present in MRDA. Under MRDA an utterance gets one general and zero or more specific tags. The general *Question* tag always serves an FLF. Other utterances can either serve an FLF or a BLF based on the specific tag assigned to them. We define a function $f_{Dimension}(DA_{specific})$ that takes the specific tag[8] as input and returns the communicative function of the utterance as given by the mapping presented in appendix A (Table 12). Algorithm 1 is used to assign the CF labels. Table 4 (the left section) gives the class distribution of the resulting CF dataset.

Table 5 gives the LLM performance on detecting the communicative function of an utterance. For comparison, the performance of a rule-based baseline is also presented. The rules consist of using common short utterance texts to determine the CF. The baseline details are presented in appendix A (Algorithm 3). LLM performs significantly worse than a naive baseline. The LLM is biased towards BLF with a higher recall of 75 compared to FLF where recall drops to 45 even though more utterances serve an FLF in the dataset.

[8]If an utterance does not have a specific tag, then the general tag becomes the specific tag as well.

6.2 Turn Management

Understanding what utterances were spoken by whom is crucial for DA classification. For example, the difference between 'Mimic' & 'Repeat' is if a speaker is repeating their utterance or someone else's. Similarly, to disambiguate the different types of floor mechanisms, the model must grasp the nuances of when a turn begins and ends.

A speaker's turn can be classified into three types: No Floor, where the speaker did not intend to take the floor (*Backchannels*) or attempts to gain the floor but is unsuccessful; New Floor, where a speaker who previously did not have the floor successfully takes it; and Floor Continuation, where the speaker who already had the floor retains it in the current utterance. We create a *Turn Management* (TM) task where each utterance is assigned one of three turn classes. This assignment is based on the current utterance's DA and the surrounding speaker IDs. Algorithm 2 gives the algorithm to assign one of the three turn labels to an utterance. Table 4 (the right section) gives the TM class distribution of the resulting dataset.

To evaluate an LLM's turn management capabilities, we prompt LLMs to predict the speaker turn label for each utterance. We also compare the model's performance with a very naive baseline that looks at speaker names and two common *Backchannel* utterances [9]. The baseline is given in appendix 4. Table 6 shows that the model performs

[9]These are *'huh'* and *'uhhuh'*.

Label	LLM			Naive Baseline		
	Precision	Recall	F1	Precision	Recall	F1
No Floor	59.08	71.40	64.66	91.93	35.92	51.67
New Floor	78.80	52.28	62.86	72.95	94.17	82.21
Floor Continuation	74.18	90.29	81.44	94.44	89.99	92.17
Macro Avg	70.63	71.28	69.61	**86.45**	**73.36**	**75.35**
Accuracy		72.72			**83.63**	

Table 6: LLaMA-70B model's performance on identifying the speaker role.

#	Speaker	Utterance	DA	AP	TM	CF	DS
1	fe046	um i can yeah i mean i i think can probably schedule ten people uh whenever.	s	1a	New Floor	FLF	Not Included
2	me010	well it's it's up to you.	s	1b	New Floor	FLF	Not Included
3	me010	i mean i i uh we don't have any huge time pressure.	e	1b+	Floor Cont	BLF	Not Included
4	me010	it's just when you have	d		Floor Cont	FLF	Not Included
5	fe046	how long will it be?	bs	2a	New Floor	FLF	Not Included
6	fe046	um i i would say maybe two weeks.	s	3a	Floor Cont	FLF	Included
7	me010	yeah.	aa	2b	New Floor	BLF	Not Included
		Current Utterance					
8	me010	oh okay.	bk	3b	Floor Cont	BLF	[6]

Table 7: Dialogue excerpt with speaker information, dialogue act (DA), adjacency pair (AP), turn management (TM), communicative function (CF), and dialogue structure (DS). The example shows how DA and AP labels get mapped to the three pre-tasks labels. The final utterance shows when a 'current utterance' gets prompted for labels from LLM, the input also contains prior utterances.

Algorithm 2 Floor Status Based on DA and Speaker Roles

Input: *DA*, *curr_spk*, *last_spk*, *next_spk*
Output: Speaker Floor Label

```
if last_spk == curr_spk then
    return "floor continuation"
end if
if DA == Backchannel then
    return "no floor"
end if
if DA == Floor Grabber then
    if curr_spk == next_spk then
        return "floor new"
    else
        return "no floor"
    end if
end if
if DA ∈ {'%'} then
    return "no floor" //interrupted utterance
end if
return "floor new"
```

worse than a simple heuristic baseline. In particular, the model struggles to accurately catch the cases where a switch in the speaker takes place or when the current speaker either fails to capture the floor or never intended to. The majority of the errors stem from the model's inability to disambiguate between short response utterances and backchannels, where the former implies the speaker has the floor while the latter doesn't. In addition, the model also struggles with short interrupted utterances.

6.3 Dialogue Structure

Adjacency pairs (AP) are defined as sequences of two utterances that are: 1) produced by different speakers & 2) ordered with a first part and a second part (Levinson, 1983). Such as a question-answer pair. Therefore, APs capture local conversation structure. To test the ability of LLMs to understand direct interactions, we devise a *Dialogue Structure* (DS) task. Since MRDA has been annotated for APs, we leverage the AP annotations for this purpose by merging overlapping APs to create a DS dataset. As shown in Table 7, utterances 1, 2, & 3 will belong to the same AP while utterance 4 will not belong to any [10].

Let a Dialogue $\mathcal{D} = \{u_0, u_1, .., u_i, ..\}$ is a list of ordered utterances. A function $f_{AP}(u_i, u_j)$ returns $True$ if u_i and u_j belong to the same AP and $False$ otherwise. For an utterance u_i, if it is part of an AP, then the DS is given as

$$DS(u_i) = [j \mid j < i, f_{AP}(u_i, u_j) == True]$$

Given an utterance u_i, the DS task is to identify the preceding utterances that are in the same AP as u_i. To prompt the models, we provide past utterances falling within a fixed window size w [11] i.e., $[u_{i-w}, u_{i-w+1}, .., u_{i-1}]$ and ask the model to out-

[10] 40% of utterances in MRDA test set belong to an AP.

[11] w is set to 10 to match the context window used for DA classification.

Model	ARI	NMI	Perfect Match
Naive Baseline	0.2668	0.340	0.17
LLaMA-3.1-70B	0.3763	0.4257	0.21

Table 8: LLM performance compared to a simple baseline that always selects the most recent utterance as relevant context.

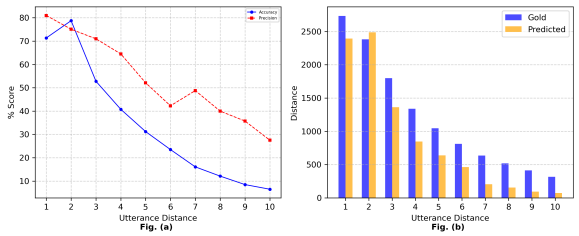

Figure 2: Utterance position is in relation to the distance of context from the current utterance. A position of 1 implies the immediately preceding utterance. (a) Context selection accuracy and precision. The model's ability to distill relevant context deteriorates as the utterances get further away. (b) The model is biased towards nearer utterances.

put the utterance numbers of those utterances that should be in the same AP as u_i. We only prompt the model for utterances that belong to an AP, so a direct interaction is guaranteed to exist.

This task can also be viewed as a clustering problem with the number of clusters fixed to two: utterances within w distance to u_i that belong to the same AP as u_i and those that do not. Therefore, we use clustering metrics to evaluate model performance. The Adjusted Rand Index (ARI) accounts for chance assignment while measuring the similarity between two clustering partitions. Normalized Mutual Information (NMI) measures how much information is shared between the predicted and gold clusters while taking the cluster size into account. The perfect match considers the clustering as correct for a data point if every utterance has been assigned to its correct cluster. This is a strict metric, as a single misplaced utterance makes the prediction incorrect. All these metrics range from 0 to 1, with 1 showing perfect agreement and 0 showing none. Table 8 compares the LLM's performance with a very simple baseline that always predicts the immediate previous utterance i.e., $DS_{baseline}(u_i) = [i-1]$. The LLM struggles to identify the local dialogue structure and performs only slightly better than the naive baseline.

Figure 2(b) shows the model's positional bias to nearer utterances; here it struggles to identify AP utterances as they move further from u_i. Similarly, Figure 2(a) shows a sharp decline in precision for more distant utterances. In other words, the model either excludes the further utterances from the AP of u_i, or when it does include them, it selects the wrong ones.

6.4 Pre-tasks and Dialogue Acts

LLMs' poor performance on the three pre-tasks suggests the models lack general dialogue understanding capabilities. The same trend also holds for smaller LLMs as well–with the performance drop on the pre-tasks proportionally more in accordance with their poor performance on DAs (Appendix C). Although we have established the importance of the pre-tasks for DA classification through a linguistic lens, this section tries to answer the question empirically.

To assess whether errors in pre-task predictions are statistically associated with DA classification errors, we employ the Chi-Square (χ^2) test. This test is well-suited for our analysis because it evaluates whether two categorical variables—pre-task correctness and DA correctness—are independent or related. We apply the test to all pre-tasks [12] separately to answer if errors on them affect DA. For all pre-tasks, we find that DA errors are not independent of pre-task predictions for $p < 0.05$, showing that errors on these pre-tasks are linked to errors for DA classification.

6.5 Fine-tuned Pre-task Baseline

To further analyze the impact of the pre-task performance on DA classification, we present the results of fine-tuned models on two of the three pre-tasks[13] (Table 9). *Turn Modeling* (He et al., 2021) uses RoBERTa to get utterance representations, followed by an LSTM to get contextual representations. It also incorporates learning two embeddings, indicating whether a speaker switch has taken place or not. As seen in Table 9, the smaller fine-tuned models can perform significantly better than the few-shot LLaMA-70B model. In addition, when turn awareness is removed (RoBERTa+BiLSTM), there is a 14-point F1 drop on 'Turn Management' task, and it also degrades the performance on DA classification. The performance on 'Communicative Function' is comparable to the model without

[12] For DS, we use the perfect match metric to identify errors.

[13] Since DS prediction cannot be easily mapped to a classification task and would require substantial architectural changes to the models.

Model	Pre-task:CF		Pre-task:TM		Dialogue Act	
	F1	Accuracy	F1	Accuracy	F1	Accuracy
Turn Modeling (He et al., 2021)	82.12	82.28	84.41	86.93	38.77	67.00
RoBERTa+BiLSTM	82.08	82.61	70.69	70.46	37.94	66.30
LLaMA-3.1-70B	55.99	58.38	69.61	72.72	19.17	31.26

Table 9: Performance of models on *Communicative Function (CF) & Turn Management (TM)* pre-tasks along with DA classification. Adding explicit turn awareness improves performance on *TM* and consequently on *DA* classification.

Model	Cohen's κ		
	TM	CF	DS
Average Human	0.7047	0.6507	0.5162
LLaMA-3.1-70B	0.4826	0.3564	0.0302
GPT-4	0.5014	0.3781	0.0251
GPT-4 Audio	0.5247	0.4097	0.0255

Table 10: Inter-annotator agreement of LLMs and Humans with the gold-derived labels. Humans achieve substantial agreement without any training, while the LLMs struggle on all tasks. *TM: Turn Management, CF: Communicative Function, DS: Dialogue Structure.*

any speaker turn awareness. This highlights that the 'Turn Modeling' model can keep track of the dialogue turn management and perform better DA classification. Therefore, designing model architectures that explicitly capture the pre-tasks can be helpful in improving DA performance.

7 Human Study

In this section, we focus on the questions of how hard these pre-tasks are and how much gap there is between human and LLM performance. We conducted a human study where two annotators–*without* going through any training–were asked to annotate three dialogues [14] for the three pre-tasks. Additional details about the annotations are present in Appendix D. To answer the first question, the annotators achieved an inter-annotator agreement (IAA) of 0.6477 for TM, 0.6653 for CF, and 0.4679 for DS as measured by Cohen's κ (Cohen, 1960). Without any training, the annotators were able to get a substantial level of agreement, which indicates that for humans these tasks are not hard.

Model	ARI	NMI	Perfect Match
Human	0.8079	0.8104	0.79
LLaMA-3.1-70B	0.1951	0.1989	0.16
GPT-4	0.1694	0.1785	0.14
GPT-4 Audio	0.1838	0.1887	0.15

Table 11: Human performance comparison with LLM performance for the dialogue structure task.

[14]There are 3433 total utterances.

To compare LLMs with human performance, we consider the gold labels as the labels for the tasks presented in section 6. The human performance is the average of both annotators. As seen in Table 10, the LLMs get a very low IAA score. Cohen's κ accounts for chance agreement and penalizes imbalanced class errors (Grandini et al., 2020). For TM and CF, while the LLMs can get a relatively higher accuracy on the task due to performing well on the 'easy' labels, they struggle with the difficult cases. In particular, the LLMs are biased to over-predicting *BLF* and *Floor Continuation*. For DS, instead of only prompting for utterances that belong to an AP, we evaluated all utterances in the test set. The model was required to either identify related preceding utterances or return an empty list when no link was present. For DS, Cohen's κ counts an utterance as correct if both the gold and model output either include it in a cluster or leave it unlinked, while Table 11 reports detailed clustering performance. The low IAA scores highlight the significant gap between LLM and human-level performance.

8 Conclusion

We study the poor performance of LLMs in multiparty DA classification. Through linguistic analysis, we identify three key pre-tasks—Turn Management, Communicative Function, and Dialogue Structure—that underpin accurate DA prediction. Our experiments demonstrate that LLMs perform poorly on these tasks, often failing to surpass naive baselines. They fail to recognize speaker roles, the broader function of an utterance, and exhibit a bias toward the most recent utterances when predicting the dialogue structure. Statistical analysis confirms a strong association between errors in these pre-tasks and DA misclassifications. A human study highlights the significant gap between LLMs and human-level dialogue understanding, reinforcing the need to train models with better discourse awareness and conversational reasoning.

Limitations

We analyze LLMs under the few-shot in-context learning setting. Additional prompt engineering (PE) techniques such as chain of thought reasoning could potentially improve LLM performance. Since the performance gap between SFT smaller models and LLMs is quite big, it would be hard to match or surpass their performance using PE alone. Therefore, we did not explore additional prompt engineering techniques.

Running all the experiments on GPT-based models cost around 800 USD. While we provide the exact prompts we used, replicating those experiments can be costly. Additionally, analysis on open-source models also requires access to a substantial amount of GPU compute resources.

Ethics Statement

LLMs are being used in real-world situations that require interactions with users. In some critical domains such as counseling chatbots, if the underlying LLM is brittle and fails to accurately comprehend dialogues, this could pose a potential risk to user well-being, lead to miscommunication, or even exacerbate existing issues.

Acknowledgements

We would like to thank the anonymous reviewers for their valuable feedback and input. We gratefully acknowledge support from National Science Foundation via the award IIS-1942918. Portions of this research were conducted with the advanced computing resources provided by Texas A&M High-Performance Research Computing.

References

Ali Ahmadvand, Jason Ingyu Choi, and Eugene Agichtein. 2019. Contextual dialogue act classification for open-domain conversational agents. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, page 1273–1276, New York, NY, USA. Association for Computing Machinery.

James Allen and Mark Core. 1997. Draft of damsl: Dialog act markup in several layers.

Gaurav Arora, Shreya Jain, and Srujana Merugu. 2024. Intent detection in the age of LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1559–1570, Miami, Florida, US. Association for Computational Linguistics.

Gemma Bel-Enguix and M Dolores Jiménez-López. 2006. Ambiguous turn-taking games in conversations. In *Actes de la 13ème conférence sur le Traitement Automatique des Langues Naturelles. Posters*, pages 398–406.

Kristy Boyer, Robert Phillips, Eun Young Ha, Michael Wallis, Mladen Vouk, and James Lester. 2009. Modeling dialogue structure with adjacency pair analysis and hidden markov models. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, pages 49–52.

David Brazil. 1997. *The communicative value of intonation in English book*. Cambridge University Press.

H Bunt, J Alexandersson, J-W CHOE, FANG Chengyu, K Hasida, V Petukhova, A Popescu-Belis, and D Traum. 2012. Iso 24617-2: 2012 language resource management–semantic annotation framework (semaf)–part 2: Dialogue acts: 2012 language resource management–semantic annotation framework (semaf)–part 2: Dialogue acts.

Harry Bunt. 1994. Context and dialogue control. *Think Quarterly*, 3(1):19–31.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Mark G Core and James Allen. 1997. Coding dialogs with the damsl annotation scheme. In *AAAI fall symposium on communicative action in humans and machines*, volume 56, pages 28–35. Boston, MA.

Rajdip Dhillon, Sonali Bhagat, Hannah Carvey, and Elizabeth Shriberg. 2004. Meeting recorder project: Dialog act labeling guide. Technical report, Citeseer.

Margherita Grandini, Enrico Bagli, and Giorgio Visani. 2020. Metrics for multi-class classification: an overview. *arXiv preprint arXiv:2008.05756*.

Zihao He, Leili Tavabi, Kristina Lerman, and Mohammad Soleymani. 2021. Speaker turn modeling for dialogue act classification. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2150–2157, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Michael Heck, Nurul Lubis, Benjamin Ruppik, Renato Vukovic, Shutong Feng, Christian Geishauser, Hsien-chin Lin, Carel van Niekerk, and Milica Gasic. 2023. ChatGPT for zero-shot dialogue state tracking: A solution or an opportunity? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 936–950, Toronto, Canada. Association for Computational Linguistics.

Thomas M Holtgraves. 2013. *Language as social action: Social psychology and language use*. Psychology Press.

Dan Jurafsky. 1997. Switchboard swbd-damsl shallow-discourse-function annotation coders manual. *Institute of Cognitive Science Technical Report*.

Dongjin Kang, Sunghwan Kim, Taeyoon Kwon, Seungjun Moon, Hyunsouk Cho, Youngjae Yu, Dongha Lee, and Jinyoung Yeo. 2024. Can large language models be good emotional supporter? mitigating preference bias on emotional support conversation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15232–15261, Bangkok, Thailand. Association for Computational Linguistics.

Taewoon Kim and Piek Vossen. 2021. Emoberta: Speaker-aware emotion recognition in conversation with roberta. *arXiv preprint arXiv:2108.12009*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Harshit Kumar, Arvind Agarwal, Riddhiman Dasgupta, and Sachindra Joshi. 2018a. Dialogue act sequence labeling using hierarchical encoder with crf. 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.

Harshit Kumar, Arvind Agarwal, Riddhiman Dasgupta, and Sachindra Joshi. 2018b. Dialogue act sequence labeling using hierarchical encoder with crf. In *Proceedings of the aaai conference on artificial intelligence*, volume 32.

Md Tahmid Rahman Laskar, Xue-Yong Fu, Cheng Chen, and Shashi Bhushan TN. 2023. Building real-world meeting summarization systems using large language models: A practical perspective. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 343–352, Singapore. Association for Computational Linguistics.

Stephen C Levinson. 1983. Pragmatics. *Cambridge UP*.

Guan-Ting Lin and Hung-yi Lee. 2024. Can LLMs understand the implication of emphasized sentences in dialogue? In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 13391–13401, Miami, Florida, USA. Association for Computational Linguistics.

Siyang Liu, Chujie Zheng, Orianna Demasi, Sahand Sabour, Yu Li, Zhou Yu, Yong Jiang, and Minlie Huang. 2021. Towards emotional support dialog systems. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3469–3483, Online. Association for Computational Linguistics.

Yang Liu, Kun Han, Zhao Tan, and Yun Lei. 2017. Using context information for dialog act classification in DNN framework. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2170–2178, Copenhagen, Denmark. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Xiang Luo, Zhiwen Tang, Jin Wang, and Xuejie Zhang. 2024. Zero-shot cross-domain dialogue state tracking via dual low-rank adaptation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5746–5765, Bangkok, Thailand. Association for Computational Linguistics.

Md Messal Monem Miah, Adarsh Pyarelal, and Ruihong Huang. 2023. Hierarchical fusion for online multimodal dialog act classification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7532–7545, Singapore. Association for Computational Linguistics.

Wenbo Pan, Qiguang Chen, Xiao Xu, Wanxiang Che, and Libo Qin. 2023. A preliminary evaluation of chatgpt for zero-shot dialogue understanding. *arXiv preprint arXiv:2304.04256*.

Andrei Popescu-Belis. 2005. Dialogue acts: One or more dimensions. *ISSCO WorkingPaper*, 62:1–46.

Ayesha Qamar, Adarsh Pyarelal, and Ruihong Huang. 2023. Who is speaking? speaker-aware multiparty dialogue act classification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10122–10135, Singapore. Association for Computational Linguistics.

Vipul Raheja and Joel Tetreault. 2019. Dialogue Act Classification with Context-Aware Self-Attention. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3727–3733, Minneapolis, Minnesota. Association for Computational Linguistics.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. Association for Computational Linguistics.

Harvey Sacks, Emanuel A Schegloff, and Gail Jefferson. 1974. A simplest systematics for the organization of turn-taking for conversation. *language*, 50(4):696–735.

John R Searle. 1969. Speech acts: An essay in the philosophy of language. *Cambridge University*.

Guokan Shang, Antoine Tixier, Michalis Vazirgiannis, and Jean-Pierre Lorré. 2020. Speaker-change aware CRF for dialogue act classification. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 450–464, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Zhouxing Shi and Minlie Huang. 2019. A deep sequential model for discourse parsing on multi-party dialogues. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7007–7014.

Elizabeth Shriberg, Raj Dhillon, Sonali Bhagat, Jeremy Ang, and Hannah Carvey. 2004. The icsi meeting recorder dialog act (mrda) corpus. In *Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue at HLT-NAACL 2004*, pages 97–100.

Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational linguistics*, 26(3):339–373.

Zhiqiang Wang, Yiran Pang, and Yanbin Lin. 2023. Large language models are zero-shot text classifiers. *arXiv preprint arXiv:2312.01044*.

Jiabao Xu, Peijie Huang, Youming Peng, Jiande Ding, Boxi Huang, and Simin Huang. 2022. Adjacency pairs-aware hierarchical attention networks for dialogue intent classification. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7622–7626. IEEE.

Piotr Żelasko, Raghavendra Pappagari, and Najim Dehak. 2021. What helps transformers recognize conversational structure? importance of context, punctuation, and labels in dialog act recognition. *Transactions of the Association for Computational Linguistics*, 9:1163–1179.

Tenggan Zhang, Xinjie Zhang, Jinming Zhao, Li Zhou, and Qin Jin. 2024. ESCoT: Towards interpretable emotional support dialogue systems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13395–13412, Bangkok, Thailand. Association for Computational Linguistics.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

Appendix

A Pre-tasks

A.1 Communicative Function

The mapping of DA labels to their respective communicative functions is given in Table 12. A rule-based baseline is given in Algorithm 3.

Algorithm 3 Baseline for communicative function

Input: $Utterance$
Output: Communicative Function Label

if $Utterance \in$ {'um', 'and um', 'but', 'so', 'well', 'uh'} **then**
  return "Other"
end if
if $Utterance \in$ {'yeah', 'okay', 'right', 'huh', 'yes', 'yep', 'oh yeah', 'oh okay', 'uhhuh', 'no', 'i see', 'oh', 'sure'} **then**
  return "Backward"
end if
return "Forward"

A.2 Turn Management

A naive baseline for predicting the turn labels is given in Algorithm 4.

Algorithm 4 Baseline for speaker turn labels

Input: $Speaker_{last}$, $Speaker_{curr}$, $Utterance$
Output: Speaker Turn Label

if $Speaker_{curr} == Speaker_{last}$ **then**
  return "floor continuation"
end if
if $Utterance \notin$ {'huh', 'uhhuh'} **then**
  return "floor new"
end if
return "no floor"

A.3 Dialogue Structure

Prompting LLM To test how well LLMs can do on this task, we prompt for every utterance that is part of an AP separately [15]. This results in 3986 data points [16] with an average cluster size of 3.009 utterances.

B Error Analysis

Figure 3 shows that for most of the classes, the DA performance is bounded by the model's ability to first detect the correct communicative function. The exceptions are 'Apology', 'Thanks', 'Exclamation', and 'Tag Question' where the model

[15] Except the first utterance in an AP cluster since it does not have any preceding utterances to be linked with

[16] This can be thought of as an *easier* version of direct interaction prediction since we only prompt for AP utterances and the model doesn't need to first identify if a direct interaction took place or not.

Dimension	Dialogue Acts
Forward Communicative Function	Statement, **Topic Change**, Y/N Question, Wh-Question, Or Question, Or Clause After Y/N Question, Open-ended Question, Rhetorical Question, Command, Suggestion, Commitment, **Follow Me**, Exclamation, Apology, Thanks, Welcome, Tag Question, Declarative Question
Backward Communicative Function	Backchannel, Acknowledgement, Assessment/Appreciation, Rhetorical Question Backchannel, Accept, Partial Accept, Affirmative Answer, Reject, Partial Reject, Dispreferred Answer, Negative Answer, Maybe, No Knowledge, Repetition Request, **Understanding Check**, **Repeat**, Mimic, Summary, Correct Misspeaking, **Self-Correct Misspeaking**, **Defending/Explanation**, Elaboration, Collaborative Completion, Downplayer, Sympathy
Communicative Status	Self Talk, Third Party Talk, Indecipherable, **Interrupted**, Abandoned, Nonspeech
Information Level	About-Task
Other	Hold, **Floor Grabber**, **Floor Holder**, **Joke**

Table 12: Mapping MRDA tags into the dimensions used in SWBD-DAMSL. Tags only present in MRDA are highlighted.

can assign the correct DA class but fails to identify the communicative function. The communicative function performance of the first three tags is almost zero–this also shows the model's bias to over-assigning the BLF. In contrast, there are some classes where the model understands the high-level function but is not able to disambiguate the specific DA. These include 'Open-ended Question' and 'Or Question', Figure 4 shows that these classes are often confused with 'Statement' or other types of question classes that have FLF.

Figure 4 and Figure 5 show the confusion matrix using LlaMA-70B and GPT4o-Audio models respectively on the MRDA testset. *Welcome* class has no instances in the testset.

C Smaller Model Performance on Pretasks

Table 13 gives the performance metrics of the smaller models on CF and TM.

D Human Study Guidelines

The annotations were done by two graduate students with computer science and linguistics background. They did not go through any training in order to properly capture the difficulty of this task. Three dialogues were chosen from the development set randomly, these are *Bed010, Bmr014, Bmr013*. The annotators were given the corrected transcript along with the audio of the dialogues. They listened to the audio and assigned labels for all tasks simultaneously for each utterance. The instructions provided for the tasks are given below. Additionally, for the *communivative function* task, they were also provided Table 12.

Communicative Function Assign each utterance to one of the following categories:

- Backward Looking: a backward functioning utterance relates to the previous discourse. For example, an utterance might answer, accept, reject, or try to correct some previous utterance or utterances.
- Forward Looking: an utterance serves as forward functioning when it constrains the future beliefs and actions of the participants and affects the discourse. For example, a question, offer, or suggestion.
- Other: Utterances where a speaker speaks to themselves or tells a joke. This category also includes 'Floor Mechanisms', short utterances where the speaker tries to gain or keep the floor.

Turn Taking The task is to assign one of three labels indicating the speaker floor status for each utterance. The definitions of the three labels are given below:

- No Floor: The speaker does not have the floor. Usually the case for backchannels or when a speaker tries to gain the floor but fails.
- Floor Continuation: When the speaker of the current utterance already had the floor and continues to keep the floor.
- Floor New: The current speaker gains the floor while they did not previously have the floor.

Dialogue Structure For the DS task, we asked the annotators to label adjacency pairs for the whole dialogue. They were provided with AP

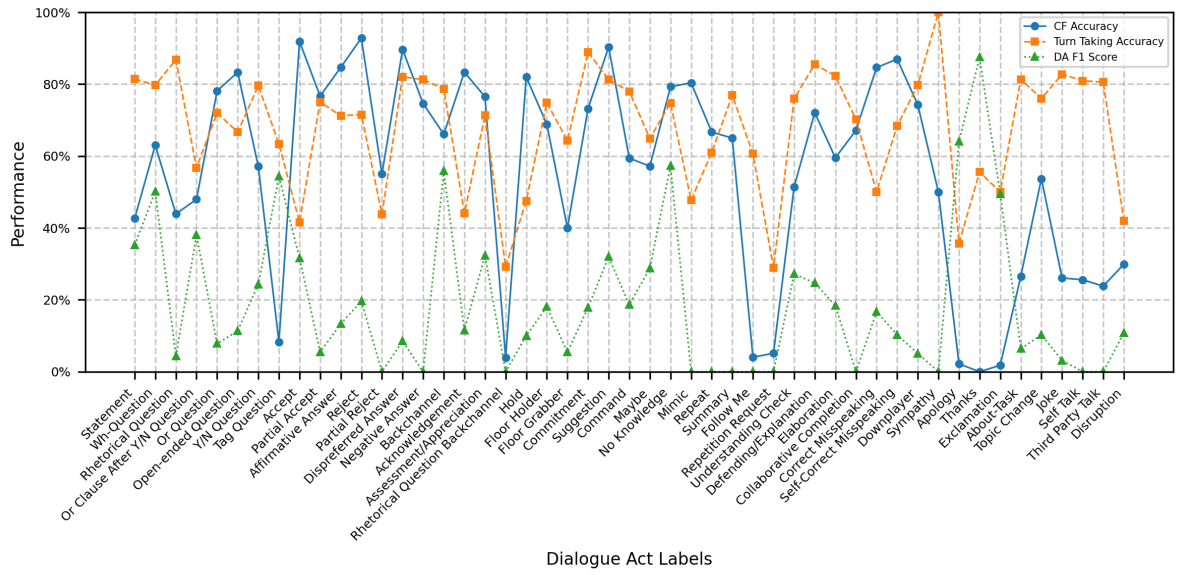

Figure 3: The green points give the F1 score on each DA class. The orange and blue points give the accuracy of turn-taking and communicative function for each DA tag. All results are from LLaMA-70B model.

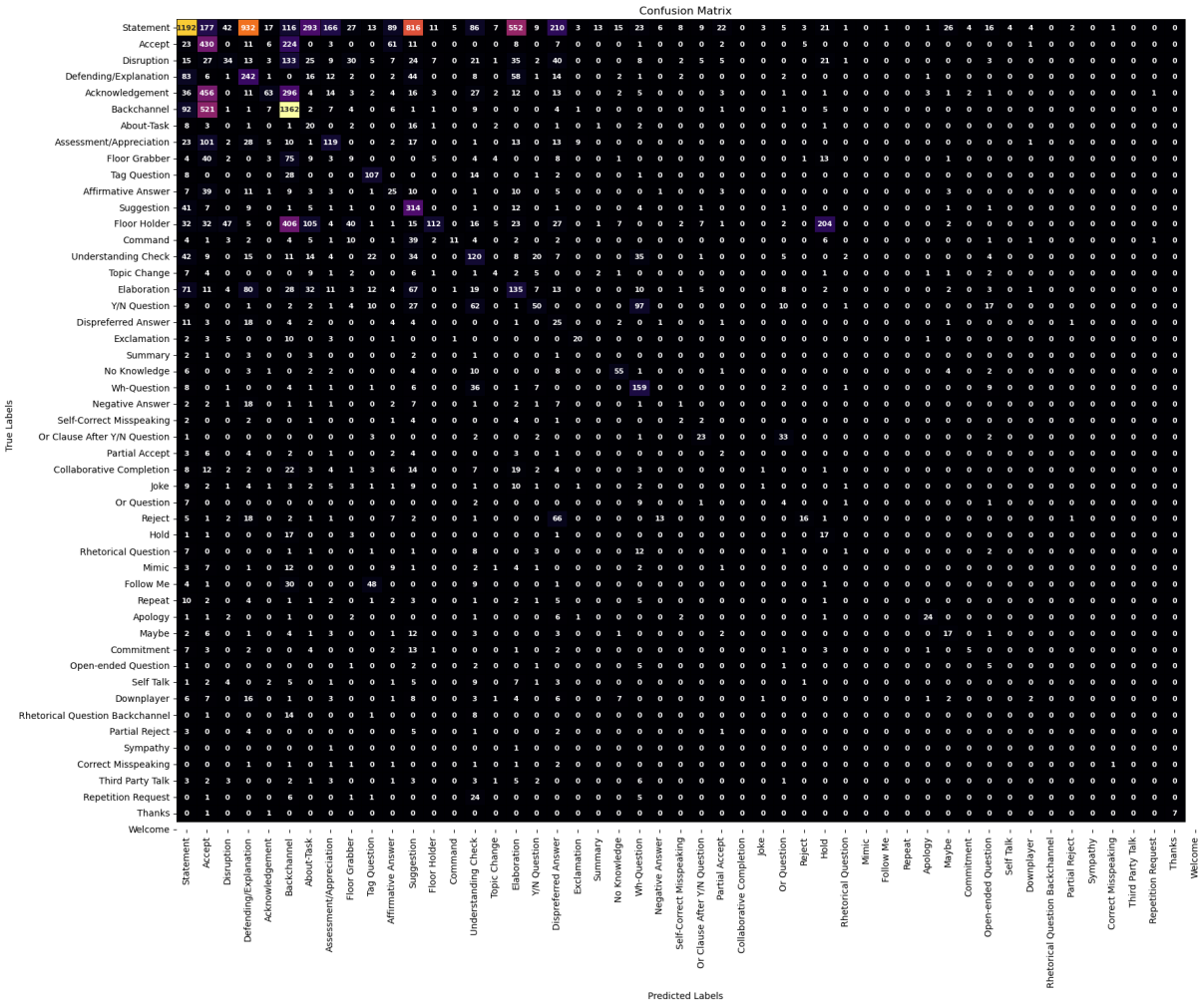

Figure 4: Confusion matrix showing the results of LLaMA-70B model on MRDA test set.

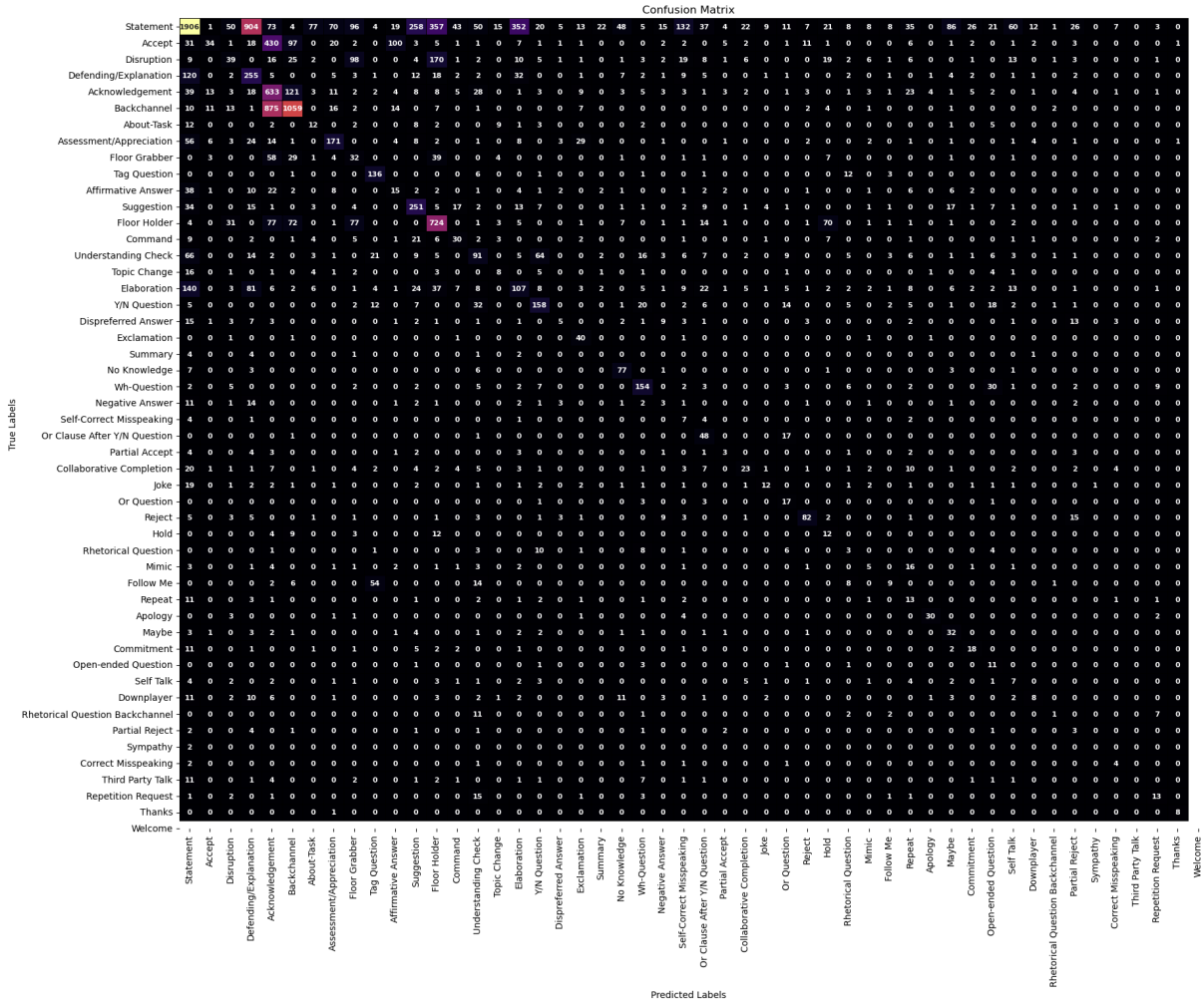

Figure 5: GPT4o-Audio model's performance on the test set of MRDA.

Model	Turn Management				Communicative Function			
	Precision	Recall	F1	Accuracy	Precision	Recall	F1	Accuracy
Gemma-7B	38.93	35.91	29.87	48.81	44.72	36.34	35.11	46.61
Mistral-7B	51.42	52.27	51.55	52.95	41.96	38.55	38.41	45.65
LLaMA-3.1-8B	57.23	48.77	45.40	56.62	46.20	50.69	46.11	47.96

Table 13: Performance comparison of smaller models on Turn Management and Communicative Function.

annotation guidelines given in the MRDA manual (Dhillon et al., 2004).

E Pre-tasks Correlation with DA Errors

To study the correlation of DA classification errors and errors on all pre-tasks, we conducted χ^2 tests. Detailed numbers are given in Table 14 below.

F Experimental Setup

The hyperparameter w that decides the number of past and future utterances to use as the context is set to 10 after using a search space of $w \in \{5, 10, 15\}$ on the development set.

Pre-task	χ^2	p-**value**
Communicative Function	141.68	1.14×10^{-32}
Turn Management	78.29	8.88×10^{-19}
Dialogue Structure	26.75	2.32×10^{-7}

Table 14: Chi-square test results for DA errors and each pre-task errors.

F.1 Models

Due to GPU memory restrictions, we use the 4-bit quantized version of LLaMA-70B model.

Model	Version	Quant.
Gemma-7B	gemma-1.1-7b-it	No
Mistral-7B	Mistral-7B-Instruct-v0.3	No
LLaMA-3.1-8B	Meta-Llama-3.1-8B-Instruct	No
LLaMA-3.1-70B	Meta-Llama-3.1-70B-Instruct	4 Bit
GPT-3.5	gpt-3.5-turbo-0125	No
GPT-4	gpt-4-turbo-2024-04-09	No
GPT-4o Audio	gpt-4o-audio-preview	No

Table 15: Model versions used in the experiments. LLaMA-3.1-70B model was quantized to 4 bit to fit on the GPUs.

F.2 Prompts

The prompts used for all the experiments are given below. Due to space constraints and readability issues, we do not include all the in-context examples used for DA classification.

Instruction
You are an intelligent annotator capable of classifying the intention behind each speaker's utterance.
You will be provided with a list of possible Dialogue Acts and their definitions.
You will be given an utterance surrounded by '#'. Your task is to predict the correct label for that utterance.
You will also be given a snapshot of the conversation to provide context for your prediction.
Return the answer in the format: 'label:predicted label'.
The Dialogue Acts and their definitions are as follows:
Statement: General statements.
Accept: A short utterance indicating acceptance of a previous speaker's statement.
Disruption: Indecipherable or disrupted speech.
Defending/Explanation: The speaker defends their opinion or provides an explanation.
Acknowledgement: Acknowledges the content of a previous speaker's utterance.
Backchannel: Indicates the listener is paying attention.
About-Task: Discusses meeting agenda or meeting direction.
Assessment/Appreciation: Expresses an evaluation or appreciation (more emotional than Acknowledgement).
Floor Grabber: The speaker, who was previously silent, attempts to gain the floor.
Tag Question: A short question following a statement to seek confirmation.
Affirmative Answer: A longer affirmative response.
Suggestion: A proposal, advice, offer, or suggestion.
Floor Holder: An utterance used mid-speech by a speaker to maintain the floor.
Command: A directive in the form of a question or statement.
Understanding Check: The speaker checks if they understood a previous speaker correctly.
Topic Change: Starts or ends a topic.
Elaboration: Expands on the speaker's own previous utterance by adding details.
Y/N Question: A yes-or-no question.
Dispreferred Answer: A direct negative response to a previous utterance.
Exclamation: Expresses excitement, surprise, or enthusiasm.
Summary: Summarizes a previous utterance or discussion.
No Knowledge: The speaker expresses a lack of knowledge.
Wh-Question: A question that requires a specific answer.
Negative Answer: An implicit negative response using hedging.
Self-Correct Misspeaking: The speaker corrects their own pronunciation or word choice.
Or Clause After Y/N Question: The speaker adds an "or" clause after a yes/no question.
Partial Accept: The speaker explicitly accepts part of a previous speaker's utterance but not all.
Collaborative Completion: The speaker attempts to complete another speaker's utterance.
Joke: A humorous or sarcastic utterance.
Or Question: A question containing two or more options.
Reject: A short negative response.
Hold: When a speaker is given the floor but delays making an utterance.
Rhetorical Question: A question to which no answer is expected.
Mimic: The speaker mimics another speaker's utterance or part of it.
Follow Me: The speaker checks if their statement is being understood.
Repeat: The speaker repeats themselves.
Apology: The speaker apologizes.
Maybe: An utterance expressing probability or uncertainty (e.g., containing the word "maybe").
Commitment: The speaker explicitly commits to a future course of action.
Open-ended Question: A question that does not seek a specific answer.
Self Talk: The speaker talks to themselves.
Downplayer: The speaker downplays or deemphasizes another utterance.
Rhetorical Question Backchannel: A rhetorical question serving as a backchannel.
Partial Reject: The speaker explicitly rejects part of another speaker's utterance.
Sympathy: An utterance expressing sympathy.
Correct Misspeaking: The speaker corrects another speaker's utterance.
Third Party Talk: Marks utterances from side conversations.
Repetition Request: The speaker asks another speaker to repeat all or part of their previous utterance.
Thanks: The speaker thanks another speaker.
Welcome: A response to an utterance marked with the "Thanks" tag.
Some examples are given below:
Example 1: Input: "Context: <Patricia>: and um I just put down some ideas. <Patricia>: you've seen some of this in the email. <Karen>: huh. <Patricia>: none of these are obligatory topics. # <Patricia>: but they're just things that I thought might be useful to discuss. # <Patricia>: just as a way of organizing the discussion. <Patricia>: but if there are other topics you'd like to discuss, that'd be great too. Now classify Utterance: <Patricia>: but they're just things that I thought might be useful to discuss.", Output: "label:Defending/Explanation".
Example 2: Input: "Context: <Richard>: We will give you an opportunity to edit all the transcripts. <Richard>: So if you say things and you don't want them to be released to the general public, you'll be given an opportunity by email to bleep out any portions. <Richard>: On the speaker form, just fill out as much of the information as you can. <Richard>: If you're not exactly sure about the region # <Richard>: we're not exactly sure either. # <Richard>: So don't worry too much about it. <Richard>: It's just self-rating. Now classify Utterance: <Richard>: we're not exactly sure either.", Output: "label:Downplayer".

Table 16: Prompt used for MRDA 50 class fine-grained DA classification. Only 2 in-context example is shown for readability.

You are a clever annotator who can understand speaker interactions in a dialogue. For a given utterance, your job is to assign one of three labels indicating the speaker floor status.
The definitions of the three labels are given below:
No Floor: The speaker does not have the floor. Usually the case for backchannels or when a speaker tries to gain the floor but fails.
Floor Continuation: When the speaker of the current utterance already had the floor and continues to keep the floor.
Floor New: The current speaker gains the floor while they did not previously have the floor.
You will be given a snapshot of a conversation context and must predict the speaker floor label of the highlighted utterance, surrounded by #, and given in Utterance.
Some examples are given below:
Example 1: Input: "Context: <Michael>: yeah they're still not decided. <James>: yeah. <Michael>: um i don't know what # <Michael>: yeah. # <Michael>: nothing much.
Now classify Utterance: <Michael>: yeah.", output: "label:floor continuation"
Example 2: Input: "Context: <Andrew>: we've got built-in downsampling. <Andrew>: and so it's only recording sixteen kilohertz data. <Andrew>: and we've got <Brian>: wait a second. # <Andrew>: we're not we're not recording the empty channels. # <Richard>: the ones that aren't filled out. Now classify Utterance: <Andrew>: we're not we're not recording the empty channels.", output: "label:floor New"
Example 3: Input: "Context: <Richard>: because Morgan said he asked you. <Patricia>: oh uh he did. <Patricia>: and and I approved. # <Richard>: uhhuh. # <Patricia>: but I think that I was uh proposed before I was asked. Now classify Utterance: <Richard>: uhhuh.", output: "label:no floor"
[Input]

Table 17: Prompt for the task of Turn Management.

Instruction
You are a clever annotator who can understand speaker interactions in a dialogue. Your job is to classify a given utterance into three categories defined below.
Backward: A backward-functioning utterance relates to the previous discourse. For example, an utterance might answer, accept, reject, or try to correct some previous utterances.
Forward: An utterance serves as forward-functioning when it constrains the future beliefs and actions of the participants and affects the discourse. For example, a question, offer, or suggestion.
Other: Utterances where a speaker speaks to themselves or tells a joke. This category also includes 'Floor Mechanisms', short utterances where the speaker tries to gain or keep the floor.
You will be given a snapshot of a conversation labeled as "Context". Predict the role of the highlighted utterance, which is surrounded by '#' and given in "Utterance".
Some examples are given below:
Example 1: Input: "Context: <Sandra>: yeah if there's anything else which we what we could add on the web site. <Sandra>: so for example if you have a small abstract or some pictures that would be fine. <Jeffrey>: okay. # <Sandra>: because now we can add some more stuff there. # <Edward>: yeah.
Now classify Utterance: <Sandra>: because now we can add some more stuff there.", output: "label:backward"
Example 2: Input: "Context: <Richard>: and and he wants to use this corpus. <Joshua>: yeah. <Joshua>: yeah exactly. # <Richard>: so # <Joshua>: i mean it's one of the areas where kemal is going to work. <Richard>: yeah.
Now classify Utterance: <Richard>: so.", output: "label:other"
Example 3: Input: "Context: <Andrew>: i don't think we do need any time aligned detail. <Andrew>: i think we just have basically one text file which runs from beginning to end. <Richard>: well if in terms of transcripts sure. <Richard>: but it might be nice to get the actual time. <Andrew>: sure. # <Andrew>: but not if it costs more. # <Richard>: right. Now classify Utterance: <Andrew>: but not if it costs more.", output: "label:backward"
Example 4: Input: "Context: <Christopher>: it depends on who else is using machines. <Christopher>: but we have more machines now. <Robert>: that's true. # <Christopher>: it's more like a day probably. # <James>: um how much worse is the short training set in terms of the performance? Utterance: <Christopher>: it's more like a day probably." output: "label:forward"
Example 5: Input: "Context: Patricia>: then you cycle through there and then you go up to this next level. <George>: so they hear all the channels at once? <Patricia>: and <Sarah>: oh okay. # <Patricia>: um # <Patricia>: let's see. <Patricia>: they hear them. Now classify Utterance: <Patricia>: um", output: "label:other"
[Input]

Table 18: Prompt for the task of Communicative Function.

Instruction
You are a clever annotator who can understand speaker interactions in a dialogue. Given a snapshot of a conversation, your job is to label the dialogue structure (DS). Dialogue Structure (DS) here is a set of utterances where at least two speakers take part in the conversation and the last utterance is functionally dependent on the previous utterance(s). Common instances include question-answer, offer-accept, direct comment pairs etc.
Give the DS predictions for the last utterance surrounded by '#' by returning the utterance number(s) of previous utterances that are part of the DS or empty list if no direct interaction is present.
Return the predictions in the format: "output:[predicted utterance numbers seperated by comma]"
Some examples are given below:
Example 1: Input: "[1] <Linda>: i don't know happy or something like that. [2] <Linda>: or it'll be a specific word. [3] <Linda>: or the type. [4] <Linda>: you'll say i need a uh spatial relation phrase here. [5] <Linda>: or i a specifier here. [6] <Linda>: uh a actual type here. [7] <Linda>: um or you could just say you know the meaning type. [8] <Linda>: so a example is the first person to do something should be an agent. [9] <Linda>: often a human. [10] <Linda>: right? # <Linda>: so if i um uh run down the street then i #", Output: "label:[]"
Example 2: Input: "[1] <James>: i mean i would suggest we discuss [2] <James>: if we're going to have a policy on it [3] <James>: that we discuss the length of time that we want to give people. [4] <Richard>: uhhuh. [5] <James>: so that we have a uniform thing. [6] <James>: so that's a month. [7] <James>: which is fine. [8] <Robert>: twelve hours. [9] <Richard>: well the only thing i said in the email is that the data is going to be released on the fifteenth. [10] <James>: i mean it seems # <Richard>: i didn't give any other deadline. #", Output: "label:[1,2,3,9]"
Example 3: Input: "[1] <James>: and there was some kind of p. make like thing that sent things out. [2] <Robert>: uhhuh. [3] <Robert>: uhhuh. [4] <Robert>: uhhuh. [5] <James>: so all twenty five people were sending things to all twenty five machines. [6] <Robert>: yeah. [7] <Robert>: yeah. [8] <James>: and and things were a lot less efficient than if you'd just use your own machine. [9] <Robert>: yep. [10] <Robert>: yeah exactly. # <Robert>: yeah you have to be a little bit careful. #", Output: "label:[10]"
Example 4: Input: "[1] <Patricia>: i mean [2] <Brian>: right. [3] <Patricia>: foot pedals i've used. [4] <Patricia>: that's fine. [5] <Patricia>: um i didn't find it any more use to me than my hand held taperecorder. [6] <Andrew>: oh. [7] <Andrew>: i see. [8] <Patricia>: however if we have a transcribing machine i would accept it. [9] <Patricia>: and having getting hold of a transcriber. [10] <Patricia>: i think that cogsci has one. # <Richard>: uhhuh. #", Output: "label:[] "'
[Input]

Table 19: LLM prompt for the task of dialogue structure.