Evaluating the Effectiveness of Large Language Models in Establishing Conversational Grounding

Anonymous ACL submission

Abstract

In this paper, we explore the concept of conversational grounding in human-machine dialogues, emphasizing its importance for effective communication, especially in spoken dialogues. Conversational grounding, vital for building dependable dialog systems, involves ensuring a mutual understanding of shared information. Despite its importance, there has been limited research on this aspect of conversation in recent years, especially after the advent of Large Language Models (LLMs). Previous studies, like those by Benotti and Blackburn (Benotti and Blackburn, 2021), highlighted the shortcomings of language models in conversational grounding but lacked a standardized benchmark for comparison. This gap in research becomes more significant considering recent advances in language models, which have led to new 'emergent' capabilities. Our study aims to evaluate the performance of Large Language Models (LLMs) in various aspects of conversational grounding, analyze why some models perform better than others, and propose ways to enhance the capabilities of the models that lag behind.

1 Introduction

The concept of "common ground" in linguistics, introduced by Clark and Brennan (Clark and Brennan, 1991), refers to the collective knowledge and assumptions that conversation participants build together. This shared understanding is not solely formed through words; it also incorporates other modalities, as highlighted by Nakano et al. (Nakano et al., 2003), such as gestures, nods, and eye contact. These non-verbal cues are crucial in creating and maintaining this common ground in face-to-face dialogues. Conversational Grounding is this process of building common grounds and involves continuous navigation, negotiation, and resolution of uncertainties. These uncertainties are often addressed by either providing additional context - for example, specifying the object "the small gate next to the bakery" or through the listener seeking clarifications, like asking, "You mean the white gate?". Since these agreements are not always explicitly expressed, participants must be adept at recognizing subtle cues of understanding from their interlocutors.

The field of conversational systems has seen various efforts aimed at addressing the challenges of grounding, particularly in the context of rule-based modular dialog systems. The pioneering work originated with (Traum and Allen, 1994), which introduced the concept of Grounding Acts (GAs). These acts serve to break down the broad process of grounding into discrete units. Although this concept offers a solid foundation for understanding and modeling conversational grounding, its application has been limited in the context of contemporary Large Language Models (LLMs). The complexity of grounding stems from the dynamic characteristics of spontaneous dialogs, which go beyond mere sequences of grammatically correct text.

In dialog systems, an effective grounding mechanism is vital for reducing ambiguities by functioning in two ways: the system can act as a speaker, adding more information if it senses confusion from the listener, or as a listener, asking for clarifications when necessary. A lack of grounding mechanism is particularly pronounced in dialogue systems that are increasingly integrating Language Models for tasks like Natural Language Understanding(NLU) and Natural Language Generation(NLG). Benotti and Blackburn (Benotti and Blackburn, 2021) had previously shown that state-of-the-art Language Models pre-trained on large amounts of conversational data like BlenderBot (Roller et al., 2020) frequently fall short in maintaining, understanding and ensuring that information has been adequately grounded by the listener during conversations. While they identified these deficiencies, their work didn't provide a comprehensive framework for evaluating different models

on their grounding capabilities. Their findings were primarily based on limited interactions with models like BlenderBot, which are less sophisticated than the more recent and advanced LLMs like Llama (Touvron et al., 2023) and GPT4 (OpenAI, 2023).

While Conversational Grounding can be a multimodal phenomenon, in this paper, we start by evaluating the performance of state-of-the-art Language Models on texts since the current dialog systems convert the speech to text before sending them as inputs to these Language Models for NLU, NLG, and in some cases the dialog management itself. The outputs are later converted into speech and gestures using separate modules. We aim to assess and enhance pre-trained LLMs' capabilities in various facets of conversational grounding due to their growing significance within the field of language models. This will lay the groundwork for more advanced modular spoken dialog systems with multimodal input and output in the future.

To this end, we have devised a series of tests to evaluate LLMs. Our approach involves analyzing the model perplexity(per token) of two potential hand-crafted responses for a given context: one being contextually appropriate and the other deceptively fitting but contextually incorrect. By comparing the perplexities of these responses, we gauge the model's proficiency in specific grounding scenarios. Our findings indicate a correlation between model performance and its size in terms of parameters. Consequently, we conduct novel tests to explore the reasons behind the underperformance of smaller models, focusing specifically on their embedding vectors. The insights gained from these investigations are then utilized to explore methods for enhancing the performance of these smaller models in conversational grounding tasks.

2 Related Works

In the field of linguistics, (Clark and Brennan, 1991) explored the inherent uncertainty present in dialogues, which interlocutors negotiate and resolve during the grounding process. Clark identified four distinct states of uncertainty: 1) B didn't notice that A uttered any utterance u. 2) B noticed that A uttered some u. 3) B correctly heard u. 4) B understood what A meant by u.

While the initial two states require work on the multimodal input and output modules of the dialog system, we focus more on the third and fourth states of uncertainty which is ensured by the LLMs in current dialog systems.

In the realm of grounding phenomena, (Traum and Allen, 1994) introduced the concept of Grounding Acts, which serves as a framework for breaking down the grounding process into its fundamental units. Within this framework, they define the following categories of GAs for every utterance:

1. **Initiate**: An initial utterance component of a grounding unit which proposes information to be grounded;
2. **Continue**: Continuation of the previous act from the same speaker;
3. **Acknowledge**: An acknowledgment of the proposed information from the interlocutor;
4. **Repair**: Correction of previously uttered material or addition of omitted material that will change the listener's interpretation of the speaker's intention.
5. **Request Repair**: Often distinguished from acknowledgment using intonation where the interlocutor asks for further clarification;
6. **Request Acknowledge**: Attempt to make the listener acknowledge the previous utterance;
7. **Cancel**: Closes off the current information without adding them to the common ground.

Subsequent theories, such as Centering Theory (Grosz et al., 1983; Barbara Grosz and Weinstein, 1986) and Domain Reference theory (Denis, 2010), introduced techniques for representing and managing grounded information. Nonetheless, their applicability was largely confined to closed domains, primarily owing to their extensive reliance on rule-based approaches for grounding. Therefore, the pursuit of more versatile models capable of categorizing utterances into diverse grounding units, regardless of the domain, holds considerable promise for advancing the field.

Similarly, other recent works have tried to focus on reference-centric multimodal Models by leveraging the success of artificial neural networks in recent times. (Fried et al., 2021) tries to solve the onecommon dataset (Udagawa and Aizawa, 2019) using an end-to-end neural network based model. However, these models look specifically at multimodal references rather than grounding.

Recent research on generative agents has highlighted the potential of Large Language Models (LLM) in interactive settings. Park et al.'s study

(Park et al., 2023) involved the creation of multiple agents, each assigned an initial identity. These agents were equipped with a memory module and relied on LLMs to assess the significance of various memories. The study demonstrated their ability to plan relevant events and execute them through human-like interactions. However, it's important to note that this research was conducted in a virtual environment with artificial agents, which does not fully replicate all human dialogue behaviors, especially conversational grounding. This limitation arises from the absence of real-time overlapping exchange of information. While this work sheds light on the potential capabilities of LLMs contrary to the results of previous work like (Benotti and Blackburn, 2021), further investigation is warranted in assessing their effectiveness in handling various grounding phenomena in natural conversations.

3 Dataset

Several datasets have been curated to support research on conversational grounding. Talk The Walk (de Vries et al., 2018) created a virtual 2D grid environment, while the HCRC Maptask (Thompson et al., 1993) had participants discuss maps and replicate routes by exchanging and negotiating their information. These conversations helped in the development of early theories and models for grounding. After assessing the existing datasets, we opted to employ the Meetup dataset (Ilinykh et al., 2019) to generate our test cases. This choice was made due to the dataset's specific design, which encourages participants to incorporate multiple instances of grounding within their conversations.

The Meetup dataset was introduced, featuring a scenario wherein two participants are placed on a 2D grid, with each vertex representing a room. The objective for the participants is to converge in the same room, despite only having visibility of their respective rooms. Navigational actions (east, west, north, or south) move participants to new rooms, unveiling the image of the newly entered room to them. Achieving the common goal necessitates the articulation of room descriptions, formulation and communication of a converging strategy, retention of room descriptions shared by the counterpart, and mental modeling of the other participant's room configurations. Although the dataset is text-based, it serves as a great resource for exploring and developing grounding models. Unlike many tasks that designate a leading role to one participant, this task creates an egalitarian dynamic where both participants can assume initiator or responder roles interchangeably. Consequently, we plan to use this dataset extensively for our experiments. The dataset contains 430 dialogs from the Meetup dataset containing 5131 utterances.

4 Models

We looked at LLMs of varying sizes and decided to test T5-Large (Raffel et al., 2020), Godel-Large (Peng et al., 2022), Llama(7 Billion)(Touvron et al., 2023), GPT 3.5(OpenAI, 2022) and GPT 4 (OpenAI, 2023). T5 is an encoder-decoder-based transformer model, while Godel, developed by Microsoft, builds upon T5 with additional fine-tuning for conversational applications; both models possess 770 million parameters. Llama and the GPT models, in contrast, are decoder-based transformer models. For T5, Godel, and Llama-7B, access to the models allowed for additional fine-tuning using next utterance prediction on the entire Meetup dataset. This enabled testing of both the original (vanilla) and fine-tuned versions of these models. It is important to note that the fine-tuned models were not exposed to the answers of the modified dialog test cases beforehand, ensuring an unbiased evaluation of their performance. Look at appendix for the finetuning training setup.

5 Perplexity Testing

Perplexity(PPL) is a measure of how well a language model predicts a sample. It quantifies the model's uncertainty in predicting a sequence of words as is given by the equation -

$$PPL(W) = e^{-\frac{1}{N}\sum_{i=1}^{N} \log_e P(w_i|w_1,\ldots,w_{i-1})} \quad (1)$$

Here, W represents the sequence of words $w_1, w_2, \ldots, w_N$, N is the length of the word sequence, and $P(w_i|w_1, \ldots, w_{i-1})$ is the probability of each word. A lower perplexity indicated a higher chance of the model generating the sequence.

In this study, we conducted an assessment of the model perplexity of candidates for the next utterances within the three conversational acts - Repair, Cancel, and Request-Repair. We separately looked at Request-Repairs that are of the Yes-No question type where the models tend to make contextual mistakes. Additionally, we examined instances of complex anaphoric references and ambiguity in references. To evaluate each phenomenon, we employed instances from the Meetup dataset by annotating the different phenomena in the dataset. Then we picked 20 instances of each phenomenon and introduced slight modifications to help us create

Instructions : Here is a conversation between two participants ..... to both participants.
Following is the dialog history along with image descriptions:
<Image A> The image showcases an oven ..... is located near the table.
[00:00:25] A: I'm in a kitchen
[00:00:43] B: In a dining room with 4 brown toys
[00:00:48] A: let me go north
<Image A> There are 4 chairs and a dining table ..... with a photo hanging on the wall.
[00:00:54] A: I see a dining room, but not your one

Figure 1: Example of input context provided to the models with the instructions, image descriptions and dialog history. Look at appendix for more complete instructions and image descriptions.

<Initial instructions + Image description>
[00:00:43] B: I am now in a dining room
[00:00:49] A: I see a library
[00:00:52] A: I'll move
[00:00:58] B: ok
[00:01:09] B: with silver latch to it
[correct] A: sorry what has a silver latch?
[wrong] A: Yes I am searching for them

Figure 2: Example of test case for a Reference Ambiguity instance for testing the perplexity

test cases. We then created a correct and a wrong response for the context and analyzed the model perplexity for them, as illustrated in Figure 2. Ideally, the perplexity of the correct response should be lower than the incorrect response.

Each input in our evaluation encompassed prior information, including instructions about the participants' situation, game rules, dialog history, and descriptions of images that the participants were viewing during the experiment as can be seen in Figure 1. These image descriptions were also interspersed within utterances during room changes. The image descriptions were initially automatically generated using the Llava model (Liu et al., 2023) and subsequently refined manually to ensure the inclusion of all pertinent information. It's worth noting that due to the unavailability of GPT3.5 and GPT4 models for direct perplexity calculation at the time of the study, we employed these models to select between the two response options as prompt as an alternative evaluation approach.

Here we provide a detailed discussion of the test case creation process for each category -

1. **Reference Ambiguity**: In case of reference ambiguity, we remove some utterances from the original dialogs to make the dialog ambiguous. Later, we test if the model is able to ask for clarifications in such cases of uncertainty as seen in Figure 2.
2. **Anaphora Reference**: We test if the model can link the anaphoric reference to the correct referent when the listener asks for clarifications. The correct response mentions the correct referent, unlike the wrong response.
3. **Repair**: In repair, we check if the model acting as a listener can correctly take the repair from the speaker into account. The correct response contains the repaired information from the interlocutor while the wrong response doesn't contain them.
4. **Cancellation**: We take dialogs where the speaker cancels the grounded information and check if the model acting as the listener can make the corrections to the grounded information. The correct response has the amended information while the wrong response doesn't.
5. **Request Repair**: We test if the model acting as a speaker can provide better repairs using the dialog context when the listener requests for a repair. The correct response provides a contextually correct repair, while the wrong response doesn't.
6. **Request Repair (Yes/No)**: For cases where the listener asks for an acknowledgment of what they have found, the questions are yes-no type. Hence, we check if the model acting as the speaker can provide the correct repair instead of a generic yes/no answer.

As can be seen from the above descriptions, we test the model in both cases where it acts as a speaker and where it acts as a listener to get a comprehensive idea of it's performance. **Note** - Look at appendix for examples of test cases for every category.

To evaluate the accuracy of our test cases, a human evaluation was conducted via Amazon Mechanical Turk. We chose the evaluators based on their past record and their location(native English speakers). We randomly selected 20% of our test cases and asked them to select the best option from the correct and wrong responses. We also provided two other options where they considered both options to be valid and neither of them to be valid. Table 1 shows that humans preferred the correct response in more than 90% of the cases. Given that each test case was independently assessed by five different individuals, the unanimous approval from this sample of our test cases affirms their validity.

Table 1: Human Evaluation of Perplexity test cases

Options	**% of times it was chosen**
Correct Option	90.65
Wrong Option	1.52
Both options are valid	6.25
None of the options are valid	1.52

D1 with repair
User A: It is overlooking the garden, with yellow seat
User B: yellow seat?
User A: sorry yellow table
User A: Do you want me to find you or you to find me?
User B: I'll look for you

D2 (paraphrased from D1 without Repair)
User A: It is overlooking the garden, with yellow table
User A: Do you want me to find you or you to find me?
User B: I'll look for you

D3 (paraphrased from D2)
User A: With a garden view, there is a yellow table
User A: Do you want me to search for you or for you to search for me?
User B: I will search for you.

D4 (with wrong information)
User A: It is overlooking the garden, with yellow seat
User A: Do you want me to search for you or for you to search for me?
User B: I will search for you.

Figure 3: Example of test case for a repair instance for understanding the hidden representations of models

6 Results - Perplexity

Table 2 reveals that smaller models like T5, Godel, and Llama(7 Billion) struggled to achieve lower perplexity for correct utterances compared to incorrect ones, indicating their limited proficiency in conversational grounding. GPT 4 on the other hand performed exceedingly well. In Table 3, the perplexity values for vanilla T5 reached as high as 10^15 showing their inability to generate the correct utterances. Contrastingly, finetuned models demonstrated significantly improved perplexity, close to 1, suggesting that finetuning aids in pattern recognition within dialogues. However, across all the categories, the smaller T5 and godel models were equally likely to generate the correct and wrong utterances as the ratio hovered around 0.5 in Table 2. Optimal model performance would have a ratio of lower perplexity for correct response close to 1 with a lower mean perplexity for correct utterances indicating that the model will actually respond with such utterances, but none of the smaller models achieved this. While Llama models performed well in asking for clarifications in case of Reference ambiguities, they were unable to ground the modified information in cases of repair and cancel and were also unable to provide repairs in cases of Request-Repair. Thus, while finetuning smaller and medium-sized models increased the likelihood of generating utterances similar to those in the dataset(like the correct and wrong responses), it does not necessarily improve the model's understanding of dialog pragmatics leading to a lack of ability to ground the conversation. In contrast, the outcomes of this experiment highlight the potential of directly employing larger models for establishing conversational grounding within our dialogue systems. the utilization of these large models may not be optimal for every dialogue system, given their increased power usage and higher cost per inference. Consequently, this prompted an investigation into the reasons behind the less effective performance of smaller and medium-sized models compared to the larger models.

7 Embedding Study

To gain deeper insights, we developed a novel method to analyze how these models process utterances at the embedding level. For this purpose, four instances of the same dialogue were generated.

1. The First instance (D1) is the original instance of a group of utterances containing the correct response of the PPL test cases of the specific phenomenon.
2. Second instance (D2) is a paraphrase of D1 without the particular phenomenon that we are testing. This is manually created keeping in mind that the overall meaning of the dialog doesn't change. A human evaluation shows that humans didn't find any difference in the meaning of the D1 and D2 as seen in Table 4. The evaluation was done similar to our previous evaluation in amazon mechanical turk where we asked them to rank the similarity between D1 and D2 on the likert scale of 1-5.
3. Third instance (D3) is a paraphrased instance of D2 where we paraphrase it utterance by utterance using GPT 4(since we are not testing GPT 4 in this test).
4. Fourth instance (D4) contains incorrect information taken from the wrong response of the PPL test cases and added to D2.

Figure 3 illustrates a test case encompassing D1, D2, D3, and D4. This test specifically examined the three GAs of Cancel, Request-Repair and Repair. Concurrently, the methodology used provided an

Table 2: Ratio of test cases where correct utterance had lower perplexity

Model	Repair	Cancel	Req-Repair(Yes/No)	Req-Repair	Anaphora	Reference Ambiguity
T5	0.45	0.55	0.65	0.50	0.45	0.35
Godel	0.40	0.65	0.45	0.50	0.35	0.40
T5 - Finetuned	0.45	0.50	0.40	0.45	0.30	0.45
Godel - Finetuned	0.35	0.50	0.45	0.45	0.40	0.45
Llama-7B	0.55	0.55	0.55	0.45	0.65	0.80
Llama-7B Finetuned	0.50	0.55	0.55	0.45	0.70	0.80
GPT 3.5	0.80	0.55	0.55	0.85	0.80	0.70
GPT 4	**0.85**	**0.95**	**1.00**	**0.95**	**0.95**	**0.85**

Table 3: Mean value of perplexity for correct utterances of each model

Model	Repair	Cancel	Request-Repair (Yes/No)	Request-Repair	Anaphora	Reference Ambiguity
T5	3.02e+15	3.46e+15	3.30e+15	2.81e+15	8.49e+14	2.00e+10
Godel	4233.29	4221.50	44379.42	44488.40	21724.60	25769.90
T5 - Finetuned	1.19	1.21	1.19	1.19	2.41	2.04
Godel - Finetuned	1.06	1.09	1.06	1.07	1.55	1.24
Llama-7B	7.12	7.10	7.75	8.00	6.93	7.70
Llama-7B Finetuned	2.91	2.91	2.89	2.92	4.72	4.63

Table 4: Human Evaluation of D1 - D2 similarity

Likert Scale	% of times it was chosen
5 (Means the same)	78.25
4 (Meaning is slightly different)	17.25
3 (Meaning is significantly different)	4.50
2 (Mean slightly opposite to each other)	0.00
1 (Mean completely opposite)	0.00

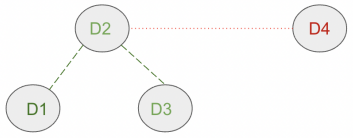

Figure 4: Pictorial representation of ideal scenario where D1 should be similar in distance to D2 as D3 and D4 should be far away

opportunity to assess implicit acknowledgments, which constitute a category of GAs. (Roque and Traum, 2008) describe Move and Use as types of implicit acknowledgments. Owing to the inherent characteristics of Reference Uncertainty, they were not examined at the embedding level in the current investigation. For Move and Use we took instances from the Meetup dataset and created the D2, D3 and D4 out of them. D4 for them was created by swapping utterances with random utterances in other dialogs making them meaningless.

The study focused on the spatial distance between the embeddings of different instances of the dialogues. Ideally, the first three dialogues (D1, D2, D3) would have close proximity in the embedding space, while D4 should be distinctly separated. This was assessed by analyzing the distances between the hidden representations of the final encoding layer of each model for each instance. Essentially, this evaluated whether the dialogue D1 containing the grounding phenomenon, bore more similarity to D3 or D4, in terms of their respective distances from D2. If the model exhibits grounding capabilities, the distance between D1 and D2 should be akin to that between D2 and D3; otherwise, it would more closely resemble the distance between D2 and D4 as depicted by Figure 4.

We created the D2 test cases from D1 for each phenomenon in the following way -

1. **Repair**: Here, we took the original dialog D1 containing the Repair and replaced the original information with the repaired information. Figure 3 provides an example where we replace the initial utterance containing 'yellow seat' with 'yellow table' directly. As a result, the information present in D1 and D2 remains the same while D4 contains 'yellow seat'.
2. **Cancel**: We remove the information that was canceled in the first place thus having the same meaning as D1.
3. **Request-Repair**: We remove the clarification asked by the listener and add the correct PPL response as repair directly in the speaker's utterance thus keeping the overall information intact. We do the same for **Request-Repair(Yes/No)**.
4. **Use**: Since a 'Use' means using the information provided in the previous utterance, we convert this implicit acknowledgment to an explicit acknowledgment to check if the model considers this acknowledgment to be similar to an explicit acknowledgment.
5. **Move**: Here, the listener moves to a new set of information thus implicitly acknowledging

Table 5: Ratio of cases where D2 is closer to D1 than D3

Model	Repair	Cancel	Req-Rep	Req-Rep(y/n)	Use	Anaphora	Move
T5	0.30	0	0.40	0.25	0.88	0.50	0.88
Godel	0.35	0	0.20	0.30	0.83	0.55	1
T5 - Finetuned	0.30	0	0.40	0.15	0.83	0.55	0.88
Godel - Finetuned	0.35	0	0.25	0.20	0.83	0.50	0.72
Llama	0.30	0.20	0.40	0.35	0.55	0.45	0.55
Llama - Finetuned	0.35	0.35	0.45	0.40	0.72	0.75	1
GPT 3.5	0.65	0.55	0.35	0.25	0.32	0.45	0.88
GPT 4	0.70	0.85	0.70	0.80	0.84	0.90	0.94

Table 6: Ratio of cases where D2 is closer to D1 than D4

Model	Repair	Cancel	Req-Rep	Req-Rep(y/n)	Use	Anaphora	Move
T5	0.35	0.50	0.40	0.40	1	0.55	1
Godel	0.35	0.65	0.25	0.50	1	0.45	1
T5 - Finetuned	0.40	0.35	0.40	0.40	0.88	0.45	1
Godel - FineTuned	0.35	0.55	0.25	0.45	0.88	0.40	0.88
Llama 7B	0.50	0.35	0.45	0.40	0.77	0.50	0.65
Llama 7B - Finetuned	0.60	0.35	0.50	0.55	0.77	0.60	0.88
GPT 3.5	0.65	0.60	0.85	0.95	0.59	0.8	0.66
GPT 4	0.85	0.95	0.70	0.95	0.91	1	1

the previous information. Thus we convert such instances to an explicit acknowledgment.

6. **Anaphora**: D2 is the dialog with the correct response in PPL testcase where the reference is correctly replaced with the object being referred while D4 has the wrong response. While D1 contains the original reference.

For GPT 3.5 and GPT 4, since we did not have the embeddings of the models, we asked the models to rank the three instances D1, D3, and D4 according to their closeness to D2. While, we do not intend to test these 2 models, we have given their performance for a benchmark.

Note - Please check the appendix for examples of each category for encoder testing.

8 Results - Embedding Study

The data presented in Table 5 reveals that smaller and medium-sized models often interpret implicit acknowledgments, such as 'Use' and 'Move', similarly to explicit acknowledgments. This suggests proficiency in these models for recognizing acknowledgments that are not explicitly expressed by participants. However, in other categories, these models perceive a closer similarity between dialogues D2 and D3 than between D1 and D3, indicating a limitation in understanding dialogs that contain nuanced grounding phenomena (D1) differently from paraphrased dialogs (D2 and D3). Table 6 further exposes a discrepancy in smaller models, where they erroneously align D4 more closely with D2 than D1 in the embedding space in 50-80% of cases. This table highlights the failure of the models at the encoder level to differentiate between utterances containing grounding phenomena and utterances containing deceptively wrong information. These distinctions, or lack thereof, in the embedding space lead to generation errors, as previously observed in our experiments. The findings from this study highlight four key insights: **1)** The model performance in differentiating between D1, D2, D3 and D4 was directly proportional to the size of the models. **2)** The models' tendency to not equate the original dialog (D1) to the paraphrased dialogs lacking the grounding phenomenon (D2 and D3), particularly for phenomena such as Repair, Request-Repair, and Cancel indicating their shortcomings in appropriately modifying information that has been corrected or canceled, or in generating contextually accurate utterances based on the recent information exchanged. In other words, **these models lack an ability to distinguish between the information presented across various temporal contexts**. **3)** The tendency of the models to confuse D2 with D4 due to word similarity, **indicating a reliance on lexical content over pragmatic understanding** as seen in Table 6. **4)** The consistent superior performance of the fine-tuned Llama model over its original version, suggests the potential benefits of further finetuning methods for enhancing model performance.

9 Positive and Negative Reward Training

Based on the embedding testing analysis, we realised that the models need to be able to distinguish at the embedding level between dialogs that sound the same but mean very different. Having found out the effectiveness of finetuning in reducing the

Table 7: Ratio of correct response having lower perplexity after positive and negative reward training

Model	Repair	Cancel	Req-Repair(Yes/No)	Req-Repair	Anaphora	Reference Ambiguity
T5	0.50	0.15	0.65	0.35	0.40	0.65
Godel	0.45	0.15	0.60	0.50	0.40	0.75
T5 - Finetuned	0.60	0.35	0.75	0.45	0.50	0.75
Godel - Finetuned	0.50	0.25	0.65	0.45	0.45	0.80
Llama-7B	0.70	0.75	0.60	0.85	0.70	0.90
Llama-7B Finetuned	0.75	0.75	0.65	0.85	0.75	0.95

Table 8: Perplexity of correct utterances for models trained with positive and negative reward

Model	Repair	Cancel	Req-Repair(Yes/No)	Req-Repair	Anaphora	Reference Ambiguity
T5	2.56e+05	6.18	20	37	4.76	1.45e+04
Godel	28.90	5.21	13.55	16.82	4.92	38.90
T5 - Finetuned	932.49	847.43	7.74e+04	1.60e+06	7617.88	5.72e+03
Godel - Finetuned	856	8.70	21.94	22.19	7.82	46.20
Llama-7B	11.88	14.63	14.32	16.51	16.26	22.31
Llama-7B Finetuned	8.95	12.93	9.07	10.67	10.43	19.90

perplexities while also helping the models distinguish more with the incorrect dialogs, we decided to create additional cases for each categories and finetune the models using Positive and Negative Reward Training (Sutton and Barto, 2018). As seen in Equation 2, this approach involved rewarding the model for correctly identifying suitable responses i.e. reduce the loss of correct response (Loss_Correct), while penalizing it for selecting incorrect utterances in the same context i.e. increase the loss of the wrong response (loss_Wrong). Both the correct and Wrong Losses are obtained using cross-entropy loss with the entire context as input and the correct and wrong responses as outputs. Here W1 and W2 are hyper-parameters empirically set as W1=4 and W2=0.5.

$$Loss = W1 * Loss\_Correct - W2 * Loss\_Wrong \quad (2)$$

Recognizing GPT 4's superior performance in our evaluations, and the need for more diverse category instances in our dataset, we utilized GPT 4 to generate 100 additional training data by feeding it examples from every category. However, it was noted that GPT 4 had limitations in creating complex cases, necessitating manual modifications to improve their quality. Tables 7 and 8 show the improvement in the performance of Llama-7B and its fine-tuned version after the positive-Negative reward training over their previous performance in Tables 2 and 3. However, the smaller models T5 and Godel were not able to improve their performance indicating a role of model size and pretraining on extensive data that leads to their ability to learn newer concepts. This indicates that a complex concept like grounding is difficult to achieve with smaller models like T5-Large even after finetuning. Conversely, a model akin to Llama's size can be trained for better grounding performance, though it may not match the proficiency of larger models like GPT-4 leading to a trade-off between better performance and computational power.

10 Conclusion and Future Work

In this study, we developed a benchmark aimed at assessing the effectiveness of LLMs in dialogue systems, with a focus on conversational grounding, and utilizing perplexity scores as a measure. Our observations revealed a direct correlation between model size and performance, highlighting the possibility of emergent properties in LLMs leading to the addressing of conversational grounding in dialogs, unlike the previous findings of (Benotti and Blackburn, 2021). Additionally, we introduced an innovative method to investigate the limited performance of smaller models by examining the embeddings from four altered versions of the same dialogue which indicated the emphasis on lexical content by smaller models. Building on these insights, we generated new training data employing positive-negative reward techniques which resulted in improved performance of medium-sized models. While they still do not match the performance level of larger models, this finding indicates that, with specific training, medium-sized models could potentially replace larger models in real-time systems where there is a need to balance performance and computational power. Future work will focus on integrating multimodal inputs and outputs in language models, recognizing their vital role in grounding processes. We also identified the necessity for improved representation techniques of grounded information to enhance storage and utilization efficiency in language models, setting a direction for subsequent research endeavors.

References

Aravind Joshi Barbara Grosz and Scott Weinstein. 1986. Towards a computational theory of discourse interpretation.

Luciana Benotti and Patrick Blackburn. 2021. Grounding as a collaborative process. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 515–531, Online. Association for Computational Linguistics.

Herbert H. Clark and Susan E. Brennan. 1991. Grounding in communication. In Lauren Resnick, Levine B., M. John, Stephanie Teasley, and D., editors, *Perspectives on Socially Shared Cognition*, pages 13–1991. American Psychological Association.

Harm de Vries, Kurt Shuster, Dhruv Batra, Devi Parikh, Jason Weston, and Douwe Kiela. 2018. Talk the walk: Navigating new york city through grounded dialogue. *ArXiv*, abs/1807.03367.

Alexandre Denis. 2010. Generating referring expressions with reference domain theory. In *Proceedings of the 6th International Natural Language Generation Conference*. Association for Computational Linguistics.

Daniel Fried, Justin Chiu, and Dan Klein. 2021. Reference-centric models for grounded collaborative dialogue. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2130–2147, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Barbara J. Grosz, Aravind K. Joshi, and Scott Weinstein. 1983. Providing a unified account of definite noun phrases in discourse. In *21st Annual Meeting of the Association for Computational Linguistics*, pages 44–50, Cambridge, Massachusetts, USA. Association for Computational Linguistics.

Nikolai Ilinykh, Sina Zarrieß, and David Schlangen. 2019. Meet up! a corpus of joint activity dialogues in a visual environment. In *Proceedings of the 23rd Workshop on the Semantics and Pragmatics of Dialogue - Full Papers*, London, United Kingdom. SEMDIAL.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning.

Yukiko Nakano, Gabe Reinstein, Tom Stocky, and Justine Cassell. 2003. Towards a model of face-to-face grounding. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 553–561, Sapporo, Japan. Association for Computational Linguistics.

OpenAI. 2022. Introducing chatgpt. `https://openai.com/blog/chatgpt`.

OpenAI. 2023. Gpt-4 technical report.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative agents: Interactive simulacra of human behavior.

Baolin Peng, Michel Galley, Pengcheng He, Chris Brockett, Lars Liden, Elnaz Nouri, Zhou Yu, Bill Dolan, and Jianfeng Gao. 2022. Godel: Large-scale pre-training for goal-directed dialog. arXiv.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric Michael Smith, Y.-Lan Boureau, and Jason Weston. 2020. Recipes for building an open-domain chatbot. In *Conference of the European Chapter of the Association for Computational Linguistics*.

Antonio Roque and David R. Traum. 2008. Degrees of grounding based on evidence of understanding. In *SIGDIAL Workshop*.

Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

Henry S. Thompson, Anne Anderson, Ellen Gurman Bard, Gwyneth Doherty-Sneddon, Alison Newlands, and Cathy Sotillo. 1993. The HCRC map task corpus: Natural dialogue for speech recognition. In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aur'elien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *ArXiv*, abs/2302.13971.

David Traum and James Allen. 1994. A "speech acts" approach to grounding in conversation.

Takuma Udagawa and Akiko Aizawa. 2019. A natural language corpus of common grounding under continuous and partially-observable context. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):7120–7127.

A Appendix

A.1 Perplexity test examples

Here we provide the remaining examples of the perplexity test. Figure 5 provides an example of the Request-Repair(Yes/No). As seen in the example, we check if the model provides a generic yes as an answer or does it check the image and figure out that it has a pink wall. We specifically check for

Request Repair(Yes/No) Perplexity Testcase Example

<Initial instructions>
<Image A> The picture depicts a calm patio with an ocean view, featuring two chairs facing the water and having pink walls. One chair is on the left and the other is positioned in the center. There is a cup on the table between them, adding warmth. A wooden railing surrounds the space for safety. Everything evokes a feeling of peace and relaxation, making it a perfect spot to spend time outside.
[00:00:57] A: I've found one. Let me know when you do.
[00:01:04] B: I am on a balcony facing an ocean
[00:01:14] A: This was thin wood railing. Two wooden folding chairs?
[00:01:19] A: You can see windows off to the left.
[00:01:26] B: yes, coffee mug on the floor?
[00:01:35] A: Yes. I think we're both in the same bedroom with a barbie theme.
[00:01:42] B: Is it the one with yellow walls?
[correct] A: No it has pink walls
[wrong] A: yes it has yellow walls

Figure 5: Example of test case for a Request-Repair(Yes/No) instance for testing the perplexity

yes/no type request repairs because the models tend to do a lot of mistakes in such cases. It is worth noting that in our test cases, the correct answer could contain a yes as well. Figure 9 shows a test case for Request-Repair where the requests are not of the yes/no type.

Figure 6 provides an example of a test case for Anaphora testing. Here we check if the model B where asked to clarify for the word 'here' is able to provide the correct referent. In some of the other test cases for anaphora, the model has to act as the listener and use the reference correctly in it's response.

Figure 7 shows an example of the repair test cases where we check the ability of the model to modify the information and ground them. Figure 8 shows an example of the cancel test case where the model has to deal with cases where the information provided by the interlocutor was canceled.

Anaphora Perplexity Testcase Example

<Initial instructions + Image descriptions>
[00:00:18] A: I am in the attick
[00:00:20] A: it is west
[00:00:42] B: I'm in the bedroom
[00:01:22] B: I see a couch here
[00:01:15] A: Sorry where do you see the couch?
[correct] B: in the bedroom
[wrong] B: in the attick

Figure 6: Example of test case for an Anaphora instance for testing the perplexity

Repair Perplexity Testcase Example

<Initial instructions + image descriptions of rooms being visited + previous utterances spoken>
User A: go north
User B: You want me to go north?
User A: sorry. I meant to go south to come inside
[correct] User B: Okay, let me go to the south
[wrong] User B: Okay, let me go to the north

Figure 7: Example of test case for a repair instance for testing the perplexity

Cancel Perplexity Testcase Example

<Initial instructions+image descriptions>
[00:00:38] A: I'm in one with diamond shelves in center
[00:00:41] A: lots of bottles
[00:00:44] A: wood racks
[00:00:54] B: I'm currently in a room with a pool table
[00:01:08] A: yellow light on ceiling
[00:01:27] B: I'm in a room with lots of bottles
[00:01:45] A: Ohh, it's not yellow
[correct] B: then what is the color of those ceilings?
[wrong] B: aah okay looking for yellow bottles then

Figure 8: Example of test case for a cancel instance for testing the perplexity

Request Repair Perplexity Testcase Example

<Initial instructions>
Below is the dialog history:
<Image B> The image is of a cluttered, tiny bedroom with two single beds pushed together, one covered in a checkered blanket. A matte black chair occupies the center and a laptop rests on one bed, a cellphone and a cup. There is a brown table containing books on top of it. The untidy room needs cleaning.
[00:00:42] B: im in the dining room
[00:00:52] A: okay describe it and I'll find you
[00:01:07] B: table with 6 chairs
[00:01:31] A: wooden walls?
[00:01:33] B: support bars on the right
[00:01:40] A: what is the color of the table?
[correct] B: It is brown in color.
[wrong] B: it is matte black in color.

Figure 9: Example of test case for a Request Repair instance for testing the perplexity

A.2 encoding test examples

Here we look at the examples of the remaining categories for the encoder testing. As seen in Figure 12 and 13, we convert the implicit acknowledgments to explicit acknowledgments to check if the model considers the both of them to be similar. In other words, does the model understand that implicit acknowledgments are also a type of an acknowledgment.

Figure 11 shows an example of Cancel where A says something but then cancels it. D2 in this case doesn't contain any information about going north. Hence, we want to check if the model is able to consider both information same or not.

Figure 10 is the same example as Figure 9 where the correct response becomes part of D2 while wrong response becomes part of D4.

A.3 Training Setup

For smaller models we used a single A100 gpu to train the models while for Llama, we used 2 gpu nodes to finetune. All the models were trained with 3 epochs. We used a Adam optimizer with a learning rate of 2e-5 and a cosine learning rate scheduler. The weight decay of the models was set at 0.01 and a batch size of 4 was used. The initial finetuning was done with a 80-20 ration of train and validation test while the entire artificial test set generated for positive-negative reward training was used for the training purpose.

Request Repair Encoding Testcase Example

D1 with Request Repair
[00:00:42] B: im in the dining room
[00:00:52] A: okay describe it and I'll find you
[00:01:07] B: table with 6 chairs
[00:01:31] A: wooden walls?
[00:01:33] B: support bars on the right
[00:01:40] A: what is the color of the table?
[00:01:46] B: It is brown in color.

D2
[00:00:42] B: im in the dining room
[00:00:52] A: okay describe it and I'll find you
[00:01:07] B: brown table with 6 chairs
[00:01:31] A: wooden walls?
[00:01:33] B: support bars on the right

D3
[00:00:42] B: im in the dining room
[00:00:52] A: okay describe it and I'll find you
[00:01:07] B: matte black table with 6 chairs
[00:01:31] A: wooden walls?
[00:01:33] B: support bars on the right

Figure 10: Example of test case for a Request Repair instance for understanding the hidden representations for each model

A.4 Complete example of instructions

Figure 14 provides the complete instruction that was provided to the models. It also shows the example of an image description that was obtained from the Llava model and later modified manually.

Cancel Encoding Testcase Example

D1 with cancel
[00:00:30] B: Okay, I got a bedroom almost all the way north
[00:00:31] A: one in a wooden cabin room, small bed
[00:00:36] B: Alright, I'll come find you
[00:00:41] A: I'm north
[00:00:53] A: no forget about it.

D2
[00:00:30] B: Okay, I got a bedroom almost all the way north
[00:00:31] A: one in a wooden cabin room, small bed
[00:00:36] B: Alright, I'll come find you

Figure 11: Example of test case for a Cancel instance for understanding the hidden representations for each model

Use Encoding Testcase Example

D1 with use
[00:00:10] B: I am in the kitchen
[00:00:21] A: Stay there and I will come. [Use]
[00:00:30] B: Okay

D2
[00:00:10] B: I am in the kitchen
[00:00:15] A: okay [explicit acknowledgment]
[00:00:21] A: Stay there and I will come.
[00:00:30] B: Okay

Figure 12: Example of test case for a Use instance for understanding the hidden representations for each model

Move On Encoding Testcase Example

D1 with use
[00:00:08] B: I am going to the left of the room
[00:00:15] A: Let me stay here
[00:00:30] B: Okay

D2
[00:00:08] B: I am going to the left of the room
[00:00:12] A: okay got it.
[00:00:15] A: Let me stay here
[00:00:30] B: Okay

Figure 13: Example of test case for a Move On instance for understanding the hidden representations for each model

Instructions : Here is a conversation between two Participants A and B who are in a virtual space that has lots of different rooms that are depicted with images. Each room has a type (such as kitchen, bathroom, bedroom, etc.). The participants are initially located in different rooms. The goal of the game is for the two participants to locate themselves in the same room. In order to achieve this goal, the participants communicate with one another by text and describe the room they find themselves in. On the basis of those descriptions, they move to different rooms and describe their new room to the other participant. The game ends when the two participants find themselves in the same room. We translated the images that the participants saw into text. That description of the room is provided below as soon as a participant enters a given room. The current room description of User A starts with a token <Image A> and the current room description of User B starts with a token <Image B>. Every utterance from A or B is preceded with a timestamp closed under brackets. Some text is provided by GM, a non-participant in the game who provides essential information regarding the game to both participants.

Following is the dialog history along with image descriptions :

<Image A> The image showcases a large, modern kitchen with dark wood cabinets and sleek black countertops. The kitchen is well-equipped with a stove top oven positioned under a ventilation fan, a microwave situated above the oven, and a refrigerator placed on the right side of the room. There are several items placed on the countertops, including a bowl, a few apples, and an orange. The kitchen also features a dining table with chairs placed around it. A potted plant adds a touch of greenery to the room, located near the dining table.

[00:00:19] B: i am currently outside

.....

Figure 14: Example of complete input context provided to the models including the instructions, image descriptions, and some dialog history