Dynamic Task-Oriented Dialogue: A Comparative Study of Llama-2 and BERT in Slot Value Generation

Tiziano Labruna and **Sofia Brenna**
Fondazione Bruno Kessler
Via Sommarive 18, Trento, Italy
Free University of Bozen-Bolzano
Piazza Università 1, Italy
tlabruna@fbk.eu, sbrenna@fbk.eu

Bernardo Magnini
Fondazione Bruno Kessler
Via Sommarive 18, Trento, Italy
magnini@fbk.eu

Abstract

Recent advancements in instruction-based language models have demonstrated exceptional performance across various natural language processing tasks. We present a comprehensive analysis of the performance of two open-source language models, BERT and Llama-2, in the context of dynamic task-oriented dialogues. Focusing on the Restaurant domain and utilizing the MultiWOZ 2.4 dataset, our investigation centers on the models' ability to generate predictions for masked slot values within text. The dynamic aspect is introduced through simulated domain changes, mirroring real-world scenarios where new slot values are incrementally added to a domain over time. This study contributes to the understanding of instruction-based models' effectiveness in dynamic natural language understanding tasks when compared to traditional language models and emphasizes the significance of open-source, reproducible models in advancing research within the academic community.

1 Introduction

In recent years, the landscape of natural language processing (NLP) has witnessed a shift towards leveraging instruction-based models, marking a departure from traditional approaches. These instruction-based models have demonstrated exceptional performance across a diverse range of complex tasks that were traditionally deemed challenging for automated solutions. Unlike closed-source linguistic models, typified by industry leaders such as OpenAI, which have dominated the market, we observe a growing interest in open-source alternatives. The inherent transparency and reproducibility of open-source models provide a conducive platform for academic research, fostering valuable experiments in diverse domains. Numerous studies have already assessed the effectiveness of open-source instruction-based models across various natural language processing (NLP) tasks.

Several notable examples include research efforts focused on fine-tuning Llama-2 for diverse applications, such as Question Answering and Text Summarization in the medical domain (Toma et al., 2023). Additionally, investigations have been conducted using OPT to generate synthetic dialogues in social contexts (Chen et al., 2023). Other studies have compared the performance of open-source models like Alpaca-Lora with proprietary alternatives, specifically in the realm of Dialogue State Tracking (Hudeček and Dušek, 2023). Furthermore, there have been assessments of Llama's performance in responding to user instructions within real-world scenarios (Ji et al., 2023).

One critical challenge in real-world applications is the dynamic nature of domains, where constant changes necessitate adaptations in dialogue systems. Previous studies (Labruna and Magnini, 2021, 2023) have shown how this domain shifts significantly deteriorate the performance of models trained on outdated data. With the emergence of instruction-based models, we aim to explore their efficacy in addressing this challenge compared to traditional models.

In this study, we focus on the task of dynamically substituting slot values for masked entities in task-oriented dialogues. This becomes particularly crucial in scenarios where domains evolve, prompting changes in slot values (e.g., a restaurant transitioning from offering "Indian" to "Italian" cuisine). We conduct a comparative analysis involving Llama-2 (Touvron et al., 2023), a state-of-the-art open-source instruction-based model, and BERT (Devlin et al., 2019), a traditional open-source language model.

Figure 1 provides an illustrative example, depicting an original dialogue with masked slot values (a), a dialogue with values generated by Llama to replace the masks (b), and a dialogue with values generated by BERT for the same task (c).

The primary contributions of this paper can be

Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop, pages 358–368

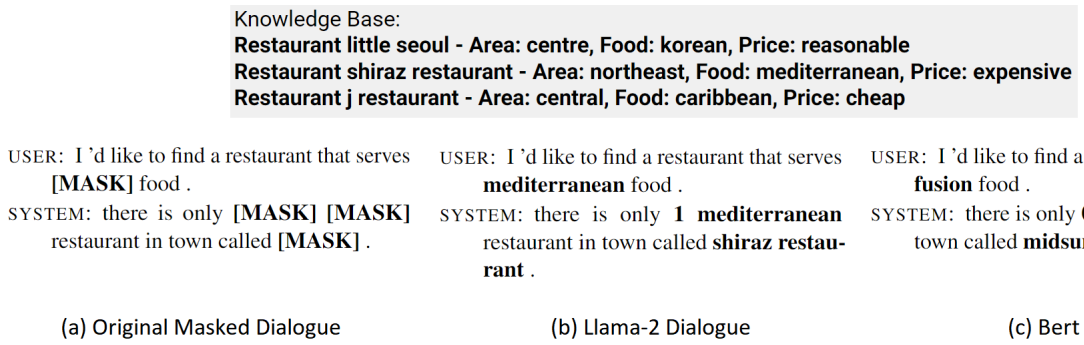

Figure 1: Comparison on the slot values substitutions made by Llama-2 and BERT when tasked with generated values to substitute masks in a task-oriented dialogue.

summarized in three key points: (i) providing comparative analysis of two popular open-source language models, Llama-2 and BERT, in the specific task of generating substitutes for masked slot values; (ii) assessing model performance in dynamic contexts, where slot values undergo changes, with insights into how well Llama-2 and BERT deal with real-world scenarios marked by evolving information; (iii) systematically investigating the impact of fine-tuning on model behavior, drawing attention to distinct strategies applied to Llama-2 and BERT, and offering valuable observations on model adaptability under varying conditions during inference.

The paper is structured as follows: Section 2 offers background insights into relevant topics discussed in the paper; Section 3 outlines the methodology employed for our task, emphasizing the nuances of the dynamic slot value generation; Section 4 details the experimental settings, specifically the introduction of domain changes; Section 5 describes the evaluation metrics utilized for a comprehensive assessment; Section 6 presents the experimental results; finally, Section 7 provides a comprehensive discussion of the findings and their implications for instruction-based and traditional language models in dynamic contexts.

2 Background

2.1 LLMs and Instruction Tuning

Large Language Models have demonstrated unparalleled ability to generate high-quality text. Among them we find for instance T5 (Raffel et al., 2020), LaMDA (Cohen et al., 2022), and BERT (Devlin et al., 2018). BERT is an encoder-only bidirectional model, having a hidden attention layer that has access to both context directions, that has been pre-trained for context-aware word representations and then fine-tuned i.e., specifically adapted for downstream tasks, along the "*pre-training and fine-tuning*" learning paradigm.

Nevertheless LMs, however large, often present misalignment with user intent. Instruction-tuned models (such as InstructGPT (Ouyang et al., 2022), LLama 2 (Touvron et al., 2023)) bring a solution to the problem, since they have been fine-tuned to be aligned with human conversational preferences in a supervised fashion on a dataset consisting of (instruction, output) pairs. Remarkable conversational abilities of the latest language models have been achieved with Instruction tuning (Wei et al., 2021; Sanh et al., 2021) and through aligning the output of the models to human preferences through Reinforcement Learning (Ng et al., 2000; Wilson et al., 2012; Todorov et al., 2012; Akrour et al., 2014; Mnih et al., 2015; Naeem et al., 2020) and prompting techniques (Liu et al., 2023). Llama-2 (Touvron et al., 2023) is an updated version of Llama-1, released in versions of 7B (the one we use), 13B, and 70B parameters, trained on a new publicly available data, with increased size of the pretraining corpus by 40% and doubled context length of the model.

2.2 MultiWOZ 2.4

MultiWOZ (Budzianowski et al., 2018) is a widely used task-oriented conversational dataset collected using the Wizard of Oz technique. It consists of over 10,000 dialogues, covering seven different domains, such as restaurant reservations and search for tourist attractions. In our experiments we employ dialogues in the Restaurant domain from version 2.4 (Ye et al., 2021). The dataset contains annotations structured in triplets: *domain* (e.g., RESTAURANT), *slot* (e.g., PRICE), and *slot-value* (e.g., EXPENSIVE).

2.3 Domain Knowledge

A task-oriented dialogue between a system and a user is considered as composed of a sequence of turns $\{t_1, t_2, ...t_n\}$ (Budzianowski et al., 2018). The system needs to retrieve a set of entities in a domain Knowledge Base (KB) satisfying the user's needs. KB is represented by a structured domain ontology O that represents entities (e.g., Restaurant, Hotel, Movie) according to a pre-defined set of slots S (e.g., Food, Area, Price, for the Restaurant domain), and values that a certain slot can take (e.g., Expensive, Moderate, Cheap, for the slot Price). On the basis of the entities defined in the domain ontology, the KB is then populated with instances of such entities.

As in much of the literature, we distinguish informable slots (e.g., Area) from requestable slots (e.g., PhoneNumber), whose values are normally queried only after a specific entity has been retrieved through the dialogue.

2.4 Domain Changes

The kind of domain change we are working with is slot-value change. This occurs every time a slot-value v used to describe an existing instance in the initial KB is changed with another slot-value (see Figure 1 for an example). This change may involve an already existing slot-value (e.g., a certain restaurant moved from INDIAN to PIZZA food, assuming that PIZZA was already used for other instances), or a new slot-value (e.g., moving from INDIAN to MEDITERRANEAN, which was never used before). The domain shift we are addressing involves alterations in slot-value pairs. This happens when a value v associated with a particular slot linked to an existing entity in the original KB, is substituted with a different slot-value (refer to Figure 1 for an example).

Such modifications could entail replacing an existing slot-value (for instance, a restaurant transitioning from being categorized as INDIAN cuisine to PIZZA, given that PIZZA was previously attributed to other entities) or introducing a wholly new slot-value (like transitioning from INDIAN to MEDITERRANEAN, a classification not previously employed).

3 Methodology

In this section, we outline the methodology employed to evaluate the performance of BERT and Llama-2 for the task of substituting slot values in a dialogue in the context of dynamic domain changes.

The primary task involves masking specific slot values in the utterances of a dialogue, both in user and system turns and assessing how well language models can generate appropriate substitutions for these masks.

3.1 Slot Values Prediction

For both BERT and Llama-2, the common task is to replace the masked slot values with appropriate generated text. The difference lies in the nature of the input provided to each model. In the case of BERT, a single sentence is passed with only one masked slot value at a time, and the model is prompted to generate the output for the substitution of that specific mask. Conversely, Llama-2 is presented with a more complex task. It is given a full instruction, consisting of a dialogue with all slot values masked, and is tasked with substituting all the masks based on the information contained in a KB provided alongside the instruction.

While the task for BERT is designed to evaluate the model's ability to generate accurate and contextually relevant responses when faced with isolated slot substitutions within a dialogue, the Llama-2 task is representative of a scenario where the model is required to assimilate information from a larger context and generate responses that need to maintain dialogue coherence across all the turns of the conversation, as well as adherence to the information of the KB.

3.2 Model Finetuning

In order to ensure that both BERT and Llama-2 comprehend the slot-value substitution task and the domain-specific information, a finetuning process is essential. However, the finetuning procedures differ significantly between the two models.

BERT's finetuning involves exposing the model to a list of utterances derived from all dialogues in the training dataset. The dataset comprises both user and system turns, and each utterance is treated as a separate training example. This is enough for the model to understand the probability distribution of word occurrences within the specific context of the dialogue.

In contrast, Llama-2's finetuning necessitates a more structured approach due to its instruction-based nature. Llama-2 requires explicit examples of instruction prompts along with their corresponding expected outputs. For the slot-value substitu-

tion task, the instruction prompt consists of a request of filling the values for all the masked slots in a full dialogue, based on the information contained in a certain number of KB instances. The model learns to generate substitutions for the masked slots based on the information contained in these instances.

3.3 Domain Changes Simulation

The simulation of domain changes is an integral part of our methodology, reflecting the dynamic nature of real-world interactions where shifts in information occur continuously. In task-oriented language understanding scenarios, models must adapt to evolving contexts, such as restaurants changing their food offerings or the introduction of new areas within a city. At inference time we want to see how changes in domain affect the performance of the models. To emulate the continuous evolution of task-specific domains, we incrementally introduce new slot values. These values substitute the original ones, reflecting changes in the characteristics of the entities within the domain.

The primary objective of introducing domain changes is to evaluate how these incremental shifts affect the quality of generated slot value produced by language models. Specifically, we aim to assess the models' ability to generate accurate responses in the presence of new slot values. By incrementally increasing the complexity of the task through the introduction of new slot values, we gain insights into the models' adaptability and their capacity to handle evolving task-oriented domains.

4 Experimental Setting

4.1 Domain Changes

In this subsection, we detail how we defined and implemented domain changes for our experiments, aiming to assess the models' adaptability to evolving task-specific domains.

We have defined four distinct domain change scenarios, each representing a different degree of alteration in the domain's information space. These scenarios correspond to 0%, 25%, 50%, 75%, and 100% of new slot values introduced into the KB. The term "new slot values" refers to information that replaces the original values associated with specific slots in the KB. The 0% of new slot values means that all the values remained as they were in the original KB, while the 100% of new values means that all the original values were substituted.

The new slot values were manually generated to ensure coherence with their respective slot names and to guarantee that they did not exist in the original KB. As an example, for the slot $Price$, which originally included the values $cheap$, $moderate$ and $expensive$, we defined the new values $affordable$, $reasonable$ and $economical$.

4.2 Finetuning

The finetuning data was derived from the training data of MultiWOZ 2.4. For each dialogue D in the dataset, we algorithmically extracted a subset of instances from the KB. This subset, denoted by I_D, represents all the instances that are referenced at least once in the dialogue D. Given I_D, we applied a certain amount of domain changes, as defined in Section 4.1, to these instances. The resulting set of instances after the domain changes is denoted by I'_D. We finally used the information of the instances I'_D to fill the slot values in the dialogue D, generating a new dialogue denoted by D'. Each dialogue D' is then used to generate the finetuning data.

As we discussed in Section 3.2, the requested format for the finetuning data differs a lot between BERT and Llama models. For BERT we simply included every utterance from the dialogue D' as part of the finetuning data. For Llama-2, the finetuning process was more complex. We masked all slot values in the dialogue D' and included the masked dialogue in the prompt, along with the KB instances I'_D correspondent to the specific dialogue. The original values from D' were included as the desired output to make the model learn the correct values for replacing the masks. A full example of a Llama prompt is shown at Appendix A.

We decided to finetune Llama-2 only on the 0% changes scenario, while for BERT we performed finetuning for all the domain changes scenarios. This resulted in the following models:

- LLAMA_KB0 - Llama-2 model finetuned on the no changes scenario
- BERT_KB0 - BERT model finetuned on the no changes scenario
- BERT_ADD25 - BERT model finetuned on the 25% of new slot values scenario
- BERT_ADD50 - BERT model finetuned on the 50% of new slot values scenario

- BERT_ADD75 - BERT model finetuned on the 75% of new slot values scenario

- BERT_ADD100 - BERT model finetuned on the 100% of new slot values scenario

The choice of finetuning Llama-2 only on the 0% changes situation reflects the specific setting of the model during inference, where it is provided with the instances containing the desired slot values for substituting the masks. In contrast, it is fundamental to finetune BERT for each change scenario in order to grasp the evolving task-specific domain. This experimental configuration also allows us to compare the performance of Llama-2 which handles progressively higher domain changes during inference, with that of BERT, which undergoes new finetuning for each distinct setting.

For finetuning Llama-2, we used "meta-llama/Llama-2-7b-chat-hf" (the 7 billion parameters version) as the base model, and made the following parameter choices: a batch size of 128, a micro-batch size of 32, three training epochs, a learning rate of 1×10^{-4}, a cutoff length of 512, a validation set size of 2000, LoRA radius (lora_r) set to 8, LoRA alpha (lora_alpha) set to 16, and a dropout rate of 0.05. For finetuning BERT, we used "bert-base-uncased" as the base model, and made the following parameter choices: a batch size of 32, three training epochs, a learning rate of 5×10^{-5} and made use of the Adam optimizer.

4.3 Inference

We assess model performance under the same domain change scenarios defined in Section 4.1 (0%, 25%, 50%, 75%, and 100% of new slot values).

We created the correspondent test-sets starting from the test-set of MultiWOZ 2.4 and applying the domain changes for each setting to the slot values, following the same procedure as outlined in Section 4.2. We then masked the slot values and asked the models to predict the correct substitutes to the masks. We conducted inference testing on each model, considering the specific finetunings and corresponding change settings: LLAMA_KB0 was tested on all five domain changes settings; each version of BERT (BERT_KB0, BERT_ADD25, BERT_ADD50, BERT_ADD75, and BERT_ADD100) was tested with the corresponding test-set matching the change setting it was finetuned on. For performing inference with the two models, we used the same versions as for finetuning as the base models, a temperature of 0.8 and a top_k of 200.

Model	Test Set	Exact Match
BERT_0	$KB0$	0.28
LLAMA_0	$KB0$	**0.49**
BERT_ADD25	$add25$	0.29
LLAMA_0	$add25$	0.40
BERT_ADD50	$add50$	0.21
LLAMA_0	$add50$	0.35
BERT_ADD75	$add75$	0.16
LLAMA_0	$add75$	0.31
BERT_ADD100	$add100$	0.17
LLAMA_0	$add100$	0.29

Table 1: Results of the exact match evaluation, determining the portion of generated slot values that correspond to the exact same value that were present in the original data.

5 Evaluation Metrics

5.1 Exact Match

This metric measures the precision of the generated values by determining if they match exactly with the original values in the test data (e.g. if the original value for the slot was "Indian", we count the generation as 1 only if it returns exactly "Indian", 0 otherwise), thus higher values indicate better performance. While it may not encompass every positive generation by the model, it ensures that every instance of an exact match is a correct generation. This metric is particularly strict and specific, setting it apart from others that offer a more nuanced perspective on data quality.

5.2 Data Quality Metrics

We employed five supplementary metrics to gain insights into various aspects of data quality. These metrics should not be considered in isolation; instead, they collectively offer perspectives on different characteristics of the quality of the generated values. In all these metrics, lower values indicate better model performance.

Out of KB Measures the number of slot values generated that do not correspond to any value present in the KB (e.g. "Caribbean" is generated, but no occurrence of this value is found in the KB).

Calculated as a ratio of such values to the total generated values.

Wrong Slots Measures the number of slot values generated that correspond to a value in the KB but are associated with a different slot name than the one in the original test data (e.g. the value "cheap" is generated as a substitution for a "Food" slot). Calculated as a ratio of such values to the total generated values.

Dialogue Incoherence Assesses the coherence of the dialogue by counting slot values that do not maintain the same substitution matches throughout the turns (e.g. first "Indian" is substituted to "Italian", then, later in the dialogue, another occurrence of "Indian" is substituted with "Chinese"). Calculated as a ratio on a subset of all generated values (the values for the first substitution matches are not eligible for this evaluation).

KB Quantifiers Misalignment Examines the adherence of quantifier slot values by identifying instances where the generated text indicates an incorrect number of instances in the KB (e.g. the system says that there are 2 "Indian" restaurants at "north", but there is none). Calculated as a ratio only on quantifier slot values.

No Output (Llama-2 Only) Measures the frequency of slot values for which no output is returned. This metric is exclusive to Llama-2 since BERT is instructed to return a value for a single MASK, ensuring some form of output. Calculated as a ratio of such values to the total slot values.

5.3 Manual Quality Evaluation

In addition to automated metrics, a manual quality evaluation was annotated on a subset of dialogues from each domain change setting. This qualitative assessment at the dialogue level annotated dialogues as either acceptable or not based on predefined criteria and provides nuanced perspective on overall performance and contextual coherence.

Dialogue Acceptability Is annotated on a subset of 100 dialogues: from the five domain change settings, 20 dialogues were drawn. Each dialogue in the subset was assessed in both the Llama-2 and BERT-completed versions. The annotation occurs at dialogue level, meaning that each dialogue was evaluated as a whole, so that one error invalidates the acceptability of the entire dialogue.

General criteria related to dialogue pragmatics such as naturalness and fluency have been complemented by more objective criteria such as: compliance with semantic and syntactic constraints, coherence across dialogue turns, consistency in referring to KB instances, adherence of quantifier slot values. There are two exceptions to these stringent conditions. The first concerns minor violations of syntactic constraints that have no effect on dialogue intelligibility (for example, "*a* affordable", "*an* sri lankan", "1 restaurant*s*"). Regarding the second point, we did not place as much emphasis on filling in the restaurant name slots with their proper nouns as we did on the other informable slots.

Dialogue Solutions Are intended as the number of instances from the given KB that provide a solution to the dialogue semantic and syntactic constraints while ensuring across-turn coherence and KB adherence. The number of possible solutions has been annotated for each dialogue to give insights on the performances of the models as the complexity of the task varies. For instance, a value of 1 solution means that using only values taken from the available KB for that particular dialogue, there would be only one configuration of slot values in the dialogue that would produce an acceptable dialogue. A value of 0 solutions means that there are no values in the KB that can be used to produce an acceptable dialogue.

6 Results

6.1 Exact Matches Results

Table 1 illustrates the percentage of exact match generations, as described in Section 5.1, for each domain change setting and both models. Notably, Llama-2 exhibits a substantial decrease in performance, dropping from 0.48 to 0.29, as new slot values are introduced. Despite this decline, Llama-2 consistently outperforms BERT in all scenarios.

6.2 Data Quality Results

Table 2 provides a comprehensive view of the evaluation metrics presented in Section 5.2. For the "Out of KB" metric, Llama-2 sees a slight decrease in performance as the domain changes increase, while BERT exhibits a slight improvement, however, BERT consistently remains considerably lower than Llama-2.

Regarding "Wrong Slots," both models demonstrate low percentages, with Llama-2 performing better in the no-change scenario but exhibiting a

Model	Test Set	Out of KB	Wrong Slots	Dialogue Incoherence	KB Quantifiers Misalignment	No Output
BERT_0	$KB0$	0.36	0.05	0.25	0.82	-
LLAMA_0	$KB0$	**0.15**	0.03	**0.05**	**0.52**	**0.32**
BERT_ADD25	$add25$	0.35	0.02	0.20	0.66	-
LLAMA_0	$add25$	0.16	0.05	0.07	0.72	0.34
BERT_ADD50	$add50$	0.33	0.04	0.19	0.76	-
LLAMA_0	$add50$	0.16	0.08	0.07	0.85	0.35
BERT_ADD75	$add75$	0.34	0.02	0.19	0.68	-
LLAMA_0	$add75$	0.18	0.10	0.08	0.95	0.38
BERT_ADD100	$add100$	0.34	**0.01**	0.15	0.73	-
LLAMA_0	$add100$	0.19	0.11	0.09	0.99	0.40

Table 2: Results of the quality of the values generated both by BERT and Llama-2 measured by different metrics.

Test Set	BERT	LLAMA-2
$KB0$	0.15	0.65
$add25$	0.15	0.70
$add50$	0.10	0.50
$add75$	0.15	**0.80**
$add100$	0.15	0.75

Table 3: Results of the manual quality evaluation, conducted over 20 dialogues per test set. Overall quality of each dialogue is considered. Results are obtained through the ratio of acceptable dialogues to the selected 20 dialogues per dataset.

decline in all other scenarios, particularly in the add100 scenario where it performs even ten times worse.

For "Dialogue Incoherence," Llama-2 consistently outperforms BERT across all scenarios, even though there is a slight decrease as new slot values are introduced.

In terms of "KB Quantifiers Misalignment," Llama-2 performs better only in the no-change scenario and then experiences a substantial decrease, reaching 99% of generated values that are not adherent to the KB.

Lastly, the "No Output" metric, applicable only to Llama-2, indicates a slight decrease in performance from 0.32 to 0.4.

6.3 Dialogue Acceptability Results

We finally present the results of the evaluation explained in Section 5.3 related to scoring the quality of the model generations at a dialogue level, through manual assessment.

Table 3 presents the outcomes of the evaluation focused on scoring the overall quality of each dialogue, considering factors such as coherence, naturalness, and informativeness. The results are represented as the ratio of acceptable dialogues to the total number of dialogues assessed in each test set.

Table 4 showcases the dialogue acceptability for BERT and Llama-2 across the five settings, taking into account the number of possible solutions for each dialogue, as explained in 5.3, with the columns labeled 0 to 3 sol indicating the number of potential solutions given the particular KB for each dialogue. Notably, 1 sol represents scenarios with a single solution, which tends to be more straightforward for the models. No examples were observed where the models successfully addressed cases with no solutions, meaning situations where, to be considered correct, the models should have generated out-of-KB values leading to zero instances in the KB. In the case of the other extreme, the scenario with three possible solutions, only Llama-2 succeeded in generating one acceptable dialogue.

7 Discussion

The results of our experiments reveal several interesting patterns and insights into the performance of instruction-based language models such Llama-2, in comparison to traditional language models like BERT. The most apparent trend is observed in Llama-2's performance, starting with a higher accuracy in the no-change scenario and gradually

Test Set	BERT				LLAMA-2			
	0 sol	1 sol	2 sol	3 sol	0 sol	1 sol	2 sol	3 sol
$KB0$	0	33%	67%	0	0	69%	31%	0
$add25$	0	**100%**	0	0	0	64%	36%	0
$add50$	0	**100%**	0	0	0	80%	20%	0
$add75$	0	67%	33%	0	0	75%	19%	6%
$add100$	0	67%	33%	0	0	73%	27%	0

Table 4: Overall dialogue acceptability in the five settings for BERT and Llama-2 related to the number of possible solutions for each dialogue: results are computed by the percentage of acceptable dialogues with *n* solutions over the number of acceptable dialogues.

declining as new slot values are introduced. This pattern underscores the vulnerability of even advanced instruction-based language models to the impact of domain changes. Despite their proficiency in certain tasks, these models struggle to maintain consistent performance in dynamic environments. Another clear observation is that Llama-2 consistently outperforms BERT when both models are fine-tuned on the same domain as the one present at inference time. This indicates that modern instruction-based models exhibit superior capabilities in the task of mask substitution when provided with the same domain information during training.

The comparison becomes more nuanced when domain changes are introduced. In most metrics, Llama-2 demonstrates significantly better performance than BERT across various scenarios, suggesting that even without specific fine-tuning, instruction-based models have better performance than traditional language models finetuned for the specific domain setting. In scenarios such as "Dialogue Incoherence", this outcome is expected, given that BERT solely replaces individual masks and lacks awareness of the dialogue's evolution, so that it becomes impossible for it to preserve coherence throughout the dialogue.

However, there are instances where BERT outperforms Llama-2, particularly for the metrics "Wrong Slots "and "KB Quantifiers Misalignment". In the case of "Wrong Slots", the constraint imposed on Llama-2 by instructions to extract values from the KB may lead to more instances of values being assigned to the wrong slot. For "KB Quantifiers Misalignment", the observed difference could be attributed to quantity values where dialogues necessitate indicating zero instances, possibly due to errors in the preceding part of the dialogue. BERT is more inclined to generate a value of zero in such cases (as in the example in Figure 1), whereas Llama tends to avoid failure examples and always say that there is at least one restaurant available.

The manual evaluation provides valuable insights that complement the quantitative results obtained from the automatic assessments. Notably, neither of the two models successfully assigns correct slot values to dialogues expecting 0 solutions in the KB. In these cases, models are expected to generate out-of-KB values; while they occasionally do, resulting in seemingly coherent dialogues, errors often manifest in subsequent turns.

It is essential to acknowledge that generating out-of-KB values is not always indicative of an error. This observation extends to the "Wrong Slot" measure, where values substituted for one slot type may correspond to another slot type yet remain acceptable within the utterance. Similarly, the metric for "Dialogue Incoherence" occasionally misclassifies cases as incorrect during automated assessment, which are instead considered correct in manual evaluation. For instance, instances where the model generates "north" after previously stating "northwest" could be technically correct, as "northwest" inherently implies "north." A similar situation arises with terms like "affordable," "economic," "moderate," and "reasonable," which may be considered synonymous but are treated as distinct values in the automated measure.

Additionally, a noteworthy observation is the significant difficulty observed in BERT's ability to generate restaurant names, which instead tends to substitute values like "it" or "that" pronouns. Overall, our study sheds light on both strengths and weaknesses of instruction-based language models like Llama-2 as compared to traditional models like

BERT, for our task.

8 Limitations

Our study has a number of limitations that should be taken into consideration when interpreting the results. Firstly, we utilized the smallest variant of Llama-2, with 7 billion parameters. It is plausible that larger versions of Llama-2 could yield improved performance.

Secondly, our experiments were conducted exclusively on the MultiWOZ 2.4 dataset, focusing specifically on the Restaurant domain and only considering informable slots. Consequently, the generalizability of our findings to other task-oriented dialogue scenarios, domains, or datasets may be limited.

Furthermore, in the case of Llama, we did not extensively explore the impact of using different prompts. The potential influence of varied prompts on performance remains an area that requires further investigation, and it is plausible that alternative prompts could lead to more favorable outcomes.

Finally, the comparison between instruction-based models and traditional language models in this study was restricted to two specific models — BERT and Llama-2 — each configured with specific parameter settings. Therefore, caution is advised when attempting to generalize these findings to a broader range of models and contexts.

9 Conclusion and Future Work

This study delved into the performance analysis of two language models, namely BERT and Llama-2, focusing on their ability to generate substitutions for masked slot values in task-oriented dialogues. The experimentation was grounded in the dynamic context of domain changes, simulating scenarios where new slot values are introduced. We used the MultiWOZ 2.4 dataset, specifically concentrating on the Restaurant domain and informable slots. Our methodology involved finetuning Llama-2 only on the zero-changes scenario, while BERT was fine-tuned for each change scenario.

Llama-2 demonstrated superior performance in the no-change scenario, emphasizing the efficacy of instruction-based models with consistent domain information, but faced a significant decline with increasing domain changes, ultimately falling below BERT for some of the considered quality features. We highlighted strengths and weaknesses of both approaches in dynamic task-oriented dialogue scenarios.

As emphasized in Section 8, there is potential for improvement by incorporating diverse models and datasets (such as SGD (Rastogi et al., 2020)). To address this, future research will delve into larger versions of Llama-2, explore alternative models, and incorporate varied datasets. Additionally, the investigation will consider the use of diverse prompts, including slot descriptions as seen in (Hudeček and Dušek, 2023), to enhance our comprehension of instruction-based models within dynamic contexts.

References

Riad Akrour, Marc Schoenauer, Michèle Sebag, and Jean-Christophe Souplet. 2014. Programming by feedback. In *International Conference on Machine Learning*, 32, pages 1503–1511. JMLR. org.

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

Maximillian Chen, Alexandros Papangelis, Chenyang Tao, Seokhwan Kim, Andy Rosenbaum, Yang Liu, Zhou Yu, and Dilek Hakkani-Tur. 2023. Places: Prompting language models for social conversation synthesis. *arXiv preprint arXiv:2302.03269*.

Aaron Daniel Cohen, Adam Roberts, Alejandra Molina, Alena Butryna, Alicia Jin, Apoorv Kulshreshtha, Ben Hutchinson, Ben Zevenbergen, Blaise Hilary Aguera-Arcas, Chung ching Chang, Claire Cui, Cosmo Du, Daniel De Freitas Adiwardana, Dehao Chen, Dmitry (Dima) Lepikhin, Ed H. Chi, Erin Hoffman-John, Heng-Tze Cheng, Hongrae Lee, Igor Krivokon, James Qin, Jamie Hall, Joe Fenton, Johnny Soraker, Kathy Meier-Hellstern, Kristen Olson, Lora Mois Aroyo, Maarten Paul Bosma, Marc Joseph Pickett, Marcelo Amorim Menegali, Marian Croak, Mark Díaz, Matthew Lamm, Maxim Krikun, Meredith Ringel Morris, Noam Shazeer, Quoc V. Le, Rachel Bernstein, Ravi Rajakumar, Ray Kurzweil, Romal Thoppilan, Steven Zheng, Taylor Bos, Toju Duke, Tulsee Doshi, Vincent Y. Zhao, Vinodkumar Prabhakaran, Will Rusch, YaGuang Li, Yanping Huang, Yanqi Zhou, Yuanzhong Xu, and Zhifeng Chen. 2022. Lamda: Language models for dialog applications. In *arXiv*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Vojtěch Hudeček and Ondřej Dušek. 2023. Are llms all you need for task-oriented dialogue? *arXiv preprint arXiv:2304.06556*.

Yunjie Ji, Yan Gong, Yong Deng, Yiping Peng, Qiang Niu, Baochang Ma, and Xiangang Li. 2023. Towards better instruction following language models for chinese: Investigating the impact of training data and evaluation. *arXiv preprint arXiv:2304.07854*.

Tiziano Labruna and Bernardo Magnini. 2021. Addressing slot-value changes in task-oriented dialogue systems through dialogue domain adaptation. In *International Conference Recent Advances In Natural Language Processing*, pages 780–789.

Tiziano Labruna and Bernardo Magnini. 2023. Addressing domain changes in task-oriented conversational agents through dialogue adaptation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 149–158.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. 2015. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533.

Muddasar Naeem, Syed Tahir Hussain Rizvi, and Antonio Coronato. 2020. A gentle introduction to reinforcement learning and its application in different fields. *IEEE access*, 8:209320–209344.

Andrew Y Ng, Stuart Russell, et al. 2000. Algorithms for inverse reinforcement learning. In *Icml*, volume 1, page 2.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

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. *Journal of Machine Learning Research*, 21(140):1–67.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:8689–8696.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2021. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*.

Emanuel Todorov, Tom Erez, and Yuval Tassa. 2012. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE.

Augustin Toma, Patrick R Lawler, Jimmy Ba, Rahul G Krishnan, Barry B Rubin, and Bo Wang. 2023. Clinical camel: An open-source expert-level medical language model with dialogue-based knowledge encoding. *arXiv preprint arXiv:2305.12031*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Aaron Wilson, Alan Fern, and Prasad Tadepalli. 2012. A bayesian approach for policy learning from trajectory preference queries. *Advances in neural information processing systems*, 25.

Fanghua Ye, Jarana Manotumruksa, and Emine Yilmaz. 2021. Multiwoz 2.4: A multi-domain task-oriented dialogue dataset with essential annotation corrections to improve state tracking evaluation. *arXiv preprint arXiv:2104.00773*.

A Appendix A: Llama-2 Prompt

Below is an instruction that describes a task, along with a Knowledge Base with the specific domain information that you should use and a Dialogue, on which you should work on. Write a response that appropriately completes the request.

Instruction:
Fill in the blanks [MASK] in the Dialogue based on the information of the Knowledge Base. Provide the filled values in the correct order, separated by commas, for every turn in the Dialogue. Separate turns with a new line. If the turn doesn't have any slot, just output the new line.

Knowledge Base:
Restaurant restaurant one seven - Area: centre, Food: british, Price: moderate
Restaurant cambridge lodge restaurant - Area: west, Food: european, Price: expensive

Dialogue:

USER: I 'm looking for a restaurant serving **[MASK]** food in any area .
SYSTEM: There are no **[MASK]** restaurants in the area .
USER: Well can I get the phone number to a **[MASK]** restaurant ?
SYSTEM: **[MASK]** serves **[MASK]** food . Their phone number is 01223 355166 . Can I help you with anything else ?
USER: That 's it , goodbye
SYSTEM: your welcome good bye

Response:
european\neuropean\nbritish\nrestaurant one seven, **british**

Upon replacing the slot values, which are generated as the output of the model, into the corresponding masks, the resulting dialogue will be as follows:

USER: I 'm looking for a restaurant serving **european** food in any area .
SYSTEM: There are no **european** restaurants in the area .
USER: Well can I get the phone number to a **british** restaurant ?
SYSTEM: **restaurant one seven** serves **british** food . Their phone number is 01223 355166 . Can I help you with anything else ?
USER: That 's it , goodbye
SYSTEM: your welcome good bye