Evaluation of bias and robustness in search and conversational systems

Abolghasemi, A.

3

CAUSE: Counterfactual Assessment of User Satisfaction Estimation in Task-Oriented Dialogue Systems

An important unexplored aspect in previous work on user satisfaction estimation for Task-Oriented Dialogue (TOD) systems is their evaluation in terms of robustness for the identification of user dissatisfaction: current benchmarks for user satisfaction estimation in TOD systems are highly skewed towards dialogues for which the user is satisfied. The effect of having a more balanced set of satisfaction labels on performance is unknown. However, balancing the data with more dissatisfactory dialogue samples requires further data collection and human annotation, which is costly and time-consuming. In this chapter, we leverage large language models (LLMs) and unlock their ability to generate satisfaction-focused counterfactual dialogues to augment the set of original dialogues of a test collection. We gather human annotations to ensure the reliability of the generated samples. We evaluate two open-source LLMs as user satisfaction estimators on our augmented collection against state-of-the-art fine-tuned models. Our experiments show that when used as few-shot user satisfaction estimators, open-source LLMs show higher robustness to the increase in the number of dissatisfaction labels in the test collection than the fine-tuned state-of-the-art models. Our results shed light on the need for data augmentation approaches for user satisfaction estimation in TOD systems. We release our aligned counterfactual dialogues, which are curated by human annotation, to facilitate further research on this topic.

3.1. INTRODUCTION

Task-oriented dialogue (TOD) systems help users complete specific tasks, e.g., booking a hotel or restaurant, through conversations [54, 155, 178, 204]. User

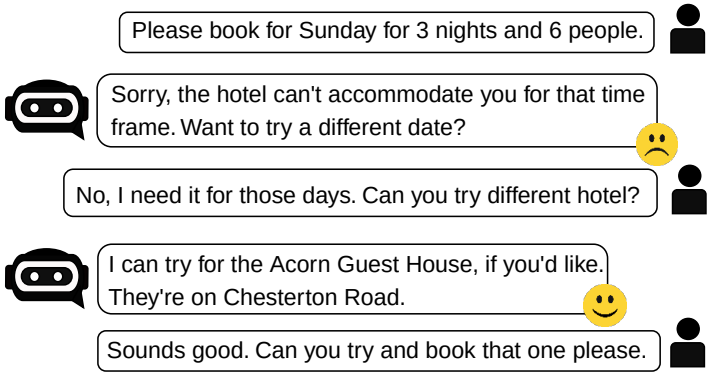

Figure 3.1: Example dialogue (snippet) between the user and the system from the MultiWOZ benchmark.

satisfaction estimation (USE) is a key task in TOD systems, aiming to measure the extent to which users are satisfied with the dialogue they are having with the system (see Figure 3.1). USE has various applications as it can be viewed as a continuous approximation of human feedback for the quality of the dialogue. Such feedback enables human intervention for users who are having a dissatisfactory dialogue with the system. Furthermore, it serves as a scalable method for the automatic evaluation of dialogue systems and helps identify and optimize a dialogue system's shortcomings [165, 194].

Prior work has studied user satisfaction estimation in TOD systems [44, 75, 172, 194] based on the user satisfaction simulation (USS) benchmark, which consists of several datasets annotated with user satisfaction labels by Sun, Zhang, Balog, Ren, Ren, Chen, and de Rijke (2021). However, the robustness of user satisfaction estimators for the identification of user dissatisfaction is an unexplored aspect in these works as most of the datasets are highly skewed towards the dialogues for which the user is satisfied. Put another way, the impact of a more balanced set of satisfaction labels on the performance of the USE models remains unknown. Nevertheless, balancing the data with more dissatisfactory dialogue samples demands further dialogue collection and human annotation which is costly and time-consuming.

To begin to address the issues raised above, we aim to expand the current imbalanced benchmarks of TOD systems with more dissatisfactory dialogues. To this aim, we leverage large language models (LLMs) and unlock their ability to generate counterfactual task-oriented dialogue samples. We use counterfactual utterance generation to generate counterpart dialogue samples with an opposite satisfaction score for a given input dialogue sample, thereby increasing the number of dissatisfaction-labeled samples in the test collections. Following the definition

of user satisfaction and the annotation guidelines from the original work in which MultiWOZ [52] and SGD [142] were annotated for user satisfaction levels,[1] we conduct human annotation on the counterfactual dialogues to ensure the quality and reliability of the generated utterances. By doing so, we introduce two augmented versions of the test collections for MultiWOZ and SGD benchmarks.

We focus on *binary* satisfaction levels, i.e., dissatisfaction and satisfaction. We argue that (i) binary labels reduce the subjectivity of annotators in labeling the dialogue, and (ii) binary satisfaction could be more relevant in some TOD system contexts, since in real-world use cases, e.g., post-hoc analysis of dialogue systems, one might only look for identification of the cases where the user is dissatisfied with the dialogue and discard the cases where the dialogue proceeds smoothly and normally. In other words, for our purposes classifying whether a dialogue is *dissatisfactory* or not is of more importance than classifying a *normal* (rating 3 in a five-point scale satisfaction levels) or *satisfying* (rate 4) from a *very satisfying* dialogue (rate 5). Table 3.1 shows both the five-point scale and the binary-level mapping of the MultiWOZ and SGD datasets used by Sun, Zhang, Balog, Ren, Ren, Chen, and de Rijke (2021). As Table 3.1 indicates, the current evaluation test collections for user satisfaction estimation in TOD systems are highly imbalanced towards the *normal* satisfaction label (3). In the binary-level satisfaction setting, this imbalance results in most dialogue samples being annotated with *satisfaction* labels, while the remaining samples are labeled as *dissatisfaction*.

Rating	MultiWOZ	SGD
1	12	5
2	725	769
3	11,141	11,515
4	669	1,494
5	6	50
Dissatisfaction	737	774
Satisfaction	11,816	13,059

Table 3.1: Data statistics of MultiWOZ and SGD on five-point and two-point satisfaction scales.

Recently, Hu, Feng, Luu, Hooi, and Lipani (2023) have shown that ChatGPT's ability to predict user satisfaction scores is comparable to that of fine-tuned state-of-the-art models. This comparable performance was only based on in-context few-shot learning (i.e., without fine-tuning) [30, 119, 135, 209]. We examine to what extent this finding on estimating user satisfaction generalizes to open-source LLMs. We use two open-source LLMs, namely, `Zephyr-7b-beta`[2] and `Mistral-7B-Instruct`[3] (to which we refer as `Zephyr` and `MistralIF`, respectively), and evaluate their

[1] We contacted the authors of [172] in which the datasets were originally annotated with satisfaction scores.

[2] https://huggingface.co/HuggingFaceH4/zephyr-7b-beta

[3] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2

performance on user satisfaction estimation on the MultiWOZ and SGD datasets.

Our experiments show that when we incorporate more dissatisfactory dialogue samples in the test collections with our methodology for generating counterfactual dissatisfying utterances, LLMs can significantly outperform the state-of-the-art fine-tuned models. We argue that this discrepancy in the performance of models across more balanced test sets is due to the imbalanced training sets with plentiful dialogue samples with satisfaction labels.

We summarize our contributions as follows:

- We show and unlock the power of LLMs in generating satisfaction-focused counterfactual dialogues in TOD systems, paving the way for data augmentation in USE for TOD systems.
- We conduct human evaluations on our generated counterfactual dialogue samples and augment the test collections of MultiWOZ and SGD benchmarks.
- Through the robustness study of USE, we find that the performance of fine-tuned state-of-the-art estimators drastically decreases with an increase in dissatisfaction-labeled dialogues in test collections.
- We show that open-source LLMs, when used in few-shot USE, maintain higher robustness in identifying user dissatisfaction in TOD systems than state-of-the-art fine-tuned estimators.

3.2. RELATED WORK

3.2.1. USER SATISFACTION ESTIMATION IN TODSS

User satisfaction estimation has been studied in the context of various information retrieval and natural language processing tasks, including conversational recommender systems [163, 164] and TOD systems [44, 131, 194]. In TOD systems, the goal of the user is to complete a specific task, e.g., booking a hotel, reserving a ticket. Depending on the flow of conversation between the user and the TOD system, user satisfaction can vary throughout the dialogue [170]. Predicting the extent to which the user is satisfied with the dialogue is defined as user satisfaction estimation. Sun, Zhang, Balog, Ren, Ren, Chen, and de Rijke (2021) study user satisfaction estimation in TOD systems and propose a benchmark for the task consisting of several datasets. They find that the core reason for user dissatisfaction is the system's failure to accurately understand the user's requests or manage their requirements effectively. Kim and Lipani (2022) propose a multi-task framework and show that user satisfaction estimation, action prediction, and utterance generation tasks can benefit from each other via positive transfer across tasks. Ye, Hu, and Yilmaz (2023) model user satisfaction across turns as an event sequence and use the dynamics in this sequence to predict user satisfaction for a current turn in the dialogue. Hu, Feng, Luu, Hooi, and Lipani (2023) leverage ChatGPT as a user satisfaction estimator and use the satisfaction scores as feedback for training a dialogue utterance generation model.

3.2.2. COUNTERFACTUAL DATA GENERATION

Generating counterfactual data samples has been studied across various natural language processing tasks [2, 118, 184, 207]. Specifically, there is a body of prior work on generating counterfactual dialogues. Li, Yavuz, Hashimoto, Li, Niu, Rajani, Yan, Zhou, and Xiong (2020) and Huang, Feng, Wu, and Du (2021) explore counterfactual dialogue generation in the context of dialogue state tracking (DTS) task. Calderon, Ben-David, Feder, and Reichart (2022) focus on the multi-label intent prediction of utterances from information-seeking dialogues and produce domain-counterfactual samples. These samples are similar to the original samples in every aspect, including the task label, yet their domain is altered to a specified one. Ben-David, Carmeli, and Anaby-Tavor (2021) study counterfactual data generation in the context of intent prediction; they address counterfactual generation, not for generating a system utterance, but for a user utterance, in contrast to the approach we take in this chapter.

There is also prior work on counterfactual data generation using LLMs, as they have shown to be highly capable in natural language generation tasks [9, 13]. For instance, Li, Xu, Miao, Zhou, and Qian (2023) explore the strengths and weaknesses of LLMs in generating counterfactual data samples. However, to the best of our knowledge, there is no prior work on satisfaction-focused counterfactual dialogue generation, which we study in this chapter.

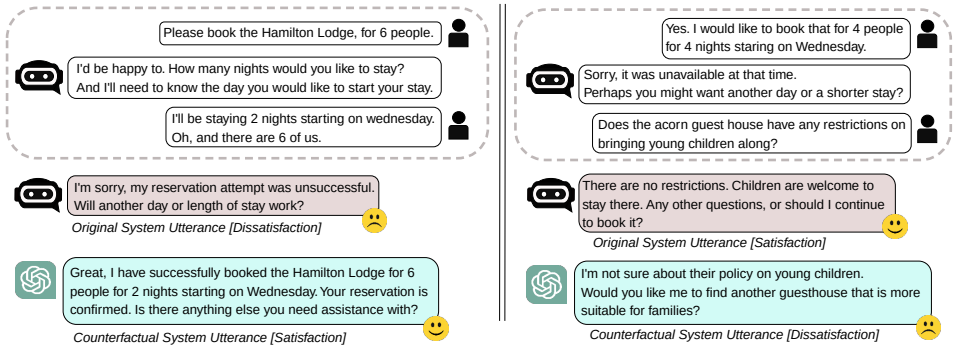

Figure 3.2: Examples of generated counterfactual system utterances. Dissatisfaction to Satisfaction (left) and vice versa (right). See Figure 3.7 in the Appendix to this chapter for the full dialogues corresponding to these examples.

3.3. USER SATISFACTION ESTIMATION

We formulate the task of user satisfaction estimation (USE) as follows. Given dialogue context $\mathscr{D}$ with T turns as $\mathscr{D} = \{(U_1, R_1), (U_2, R_2), \ldots, (U_T, R_T)\}$, where U_t and R_t stand for the t-th user utterance and system response, respectively, the goal is to estimate the user satisfaction s at the turn T. Therefore, the task objective is to learn a prediction model $P(s_T|\mathscr{D})$, where s_T is the user satisfaction at the T-th turn.

3.4. METHODOLOGY

3.4.1. COUNTERFACTUAL UTTERANCE GENERATION

Annotated dialogues with user satisfaction labels are not necessary available upon deploying TOD systems. Moreover, obtaining annotations with user satisfaction labels is both expensive and labor-intensive. However, LLMs have enabled quality text generation across various tasks [28, 171, 197, 198]. We take advantage of these models in order to generate new dialogue samples with a presumed satisfaction label in order to make up for the imbalance that exists in the benchmarks used for the evaluation of user satisfaction estimation.

Utterance Generation Task Formulation. Given a dialogue context $\mathscr{D} = \{(U_1, R_1), (U_2, R_2), \ldots, (U_T, R_T)\}$ with T turns, the goal is to generate $\widehat{R}_T$ in order to obtain $\widehat{\mathscr{D}} = \{(U_1, R_1), (U_2, R_2), \ldots, (U_T, \widehat{R}_T)\}$, where the user satisfaction label for the T-th turn for dialogue $\widehat{\mathscr{D}}$ is the opposite of user satisfaction label for $\mathscr{D}$. Our definition of counterfactual utterance is based on the annotation guidelines in [172], in which MultiWOZ and SGD with user satisfaction labels are introduced.

In order to generate a counterfactual response $\widehat{R}_T$ for a given system response R, we use few-shot in-context learning (ICL) with LLMs [30, 135]. Here, we provide the LLM `GPT-4` with an instruction regarding what a counterfactual system utterance means. We do that both when we have a satisfaction-labeled dialogue sample or a dissatifaction-labeled one. Figure 3.6 in the Appendix shows the prompt used for generating counterfactual system utterances using `GPT-4`. Clearly, we perform the generation in a *dialogue-aware* manner, i.e., the generation of counterfactual system utterance $\widehat{\mathscr{R}}$ is conditioned on the history of the dialogue between the user and the system.

Figure 3.2 shows two samples of counterfactual utterance generation. As the figure (left) shows, the counterfactual generation process is context-aware, meaning that the generated counterfactual system utterance includes information from the previous turns (i.e., context) of dialogue.

3.4.2. USER SATISFACTION ESTIMATION USING LLMS

Enabling zero-shot/few-shot (in-context learning) user satisfaction estimation could be of great use for the development and evaluation of dialogue systems. Such an in-context learning setup for the inference of user satisfaction labels facilitates the deployment of such systems as zero-shot/few-shot learning and removes the need for training samples which are costly to obtain. For instance, Hu, Feng, Luu, Hooi, and Lipani (2023) show that ChatGPT can provide a comparable performance to supervised methods. They employ ChatGPT as a user simulator to obtain user feedback on the generated utterances. While Hu, Feng, Luu, Hooi, and Lipani (2023) use zero-shot/few-shot in-context learning with a proprietary language model for user satisfaction estimation, we evaluate the performance of open-source models.

Few-shot In-context Learning. In order to estimate user satisfaction for a given dialogue, we use few-shot in-context learning [30, 135]. Figure 3.3 shows the prompt used for estimating user satisfaction using few-shot in-context learning with the two LLMs `Zephyr` [175] and `MistralIF` [84].

```
Instruction:
We want to label the user satisfaction for example dialogues. The
description of 2 labels is as follows:

"Dissatisfied": The system fails to understand or fulfill user's
request in any way.

"Satisfied": The system understands users request and either
"partially" or "fully" satisfies the request or provides information
on how the request can be fulfilled.

Example 1:
{Example Dialogue 1}
Label of Example 1 is "Satisfied".

Example 2:
{Example Dialogue 2}
Label of Example 2 is "Dissatisfied".

Example 3:
{Input Dialogue}
Label of Example 3 is:
```

Figure 3.3: The input used as the prompt for LLMs in order to predict the user satisfaction label.

3.5. EXPERIMENTAL SETUP

3.5.1. BENCHMARKS

We evaluate the models on the Multi-Domain Wizard-of-Oz (MultiWOZ) [52] and Schema Guided Dialogue (SGD) [142] benchmarks in our experimental setup. MultiWOZ and SGD are two commonly-used multi-domain task-oriented dialogue datasets and were initially annotated with user satisfaction scores by Sun, Zhang, Balog, Ren, Ren, Chen, and de Rijke (2021). We leverage the data splits used in prior work [44, 194]. Table 3.2 shows the statistics of train/validation/test splits in the MultiWOZ and SGD benchmarks.

	MultiWOZ			SGD		
Label	Train	Valid.	Test	Train	Valid.	Test
#Satisfaction	6315	775	811	6985	848	848
#Dissatisfaction	431	65	40	492	67	76
#Total	6746	840	851	7477	915	924

Table 3.2: Statistics of train/validation/test sets for the original test samples.

We also note that in this chapter we only work on turn-level satisfaction labeling. Generating a counterfactual sample for a complete dialogue requires more stratified

and complicated dialogue generation methods that are beyond the scope of this chapter.

3.5.2. EVALUATION METRICS

Following [75, 172, 194], we use Accuracy, Precision (the proportion of the predicted correct labels over the number of predicted labels), Recall (the proportion of the predicted correct labels over the number of actual labels), and the F1-score (the harmonic mean of precision and recall) as our evaluation metrics.

3.5.3. BASELINES

BERT. BERT [45] is a widely-used baseline as satisfaction label classifier in prior work [44, 92, 172, 194]. BERT achieves state-of-the-art performance in [172] and Hu, Feng, Luu, Hooi, and Lipani (2023) shows that it outperforms ChatGPT in few-shot setting. We replicate the implementation from [172] for this baseline. In addition, we up-sample the dissatisfaction class by orders of 10x up to 50x and include the models with the best and the second best performance in our results.

ASAP. ASAP is our second baseline for the evaluation against LLMs for user satisfaction estimation. Ye, Hu, and Yilmaz (2023) propose ASAP as user satisfaction estimator in which they leverage Hawkes processes [116] to capture the dynamics of user satisfaction across turns within a dialogue. Ye, Hu, and Yilmaz (2023) show that ASAP achieves state-of-the-art performance over a variety of baselines. We conduct the same aforementioned up-sampling approach of BERT for ASAP.

3.5.4. HUMAN ANNOTATION

To evaluate the quality of the generated counterfactual dialogues we conduct human evaluation on the samples for both MultiWOZ and SGD benchmarks. We use two human annotators (and a third in the case of disagreement) and annotate the counterfactual dialogues in terms of "user satisfaction," and "dialogue coherence."

Dialogue Coherence (DC). DC refers to the degree to which a generated counterfactual is relevant (fitting) to the previous turns in the dialogue, i.e., if the counterfactual system utterance is coherent with the dialogue history. An example of a non-coherent counterfactual system utterance is a case where the system answers a request for booking a hotel in a city with a response regarding the reservation of a restaurant in that city.

User Satisfaction Labeling. In the counterfactual dialogues, we only replace the last system utterance with a counterfactual one. To verify the effect of this change, we ask our annotators to label the whole dialogue in terms of user satisfaction. In the annotation pool, we mix the counterfactual dialogues with actual dialogues to prevent any learning bias. We use the same guidelines as Sun, Zhang, Balog, Ren, Ren, Chen, and de Rijke (2021) with a slight difference where we exchange the five-point scale rating with a binary-level satisfaction rating. We also note that, following Sun, Zhang, Balog, Ren, Ren, Chen, and de Rijke (2021), we use before-utterance (BU) prediction of user satisfaction scores [92]. In this approach,

user satisfaction is estimated after a system utterance and before the next user utterance. This is in contrast to after-utterance (AU) prediction [25, 31], in which the satisfaction score prediction is conducted after each user utterance, and therefore, user expressions in their utterance can be used as an indicator of their satisfaction level. While being more difficult, BU prediction enables the dialogue system to prevent potential negative user experiences by steering the conversation away from directions that might lead to dissatisfaction [92].

3.6. EXPERIMENTAL RESULTS

3.6.1. DATA QUALITY

We first assess the quality of the data that we have collected. We measure the inter-annotator agreement (IAA) between our annotators. Table 3.3 shows the agreement between the annotators on the satisfaction labels measure by Cohen's Kappa. As for DC, most of the data falls into one category (agreement on the coherence of the generated system utterance), making Kappa not a reliable metric. Instead, we use Percent Agreement which is the percentage of agreement between the two annotators.

	MultiWOZ	SGD
Dialogue Coherence (PA)	97.6	95.2
Satisfaction Label (κ)	0.84	0.86

Table 3.3: Inter-annotator Agreement (IAA) results between the two initial annotators. Percent Agreement (PA) and Cohen's Kappa (κ) are respectively used for dialogue coherence and satisfaction labels from expert annotators.

Additionally, Table 3.4 shows the ratio of correctly flipping the satisfaction status of the last system utterance, which we refer to as Counter Satisfaction Status (CSS). As the overall CSS values show, not all generated system utterances are satisfaction-focused counterfactuals of the original system utterances, i.e., 63.8 success rate for MultiWOZ and 80.3 for SGD. We only keep the samples in the CF set that are confirmed to be counterfactual by the human annotators.

Moreover, from the user evaluation in Table 3.4 we infer that `GPT-4` is better at generating dissatisfying system utterances (the CSS values in the *Satisfaction* row in Table 3.4) than at generating satisfying system utterances (the CSS values in the *Dissatisfaction* row).

Based on the labeling obtained using the three annotators, Table 3.5 shows the number of test samples for both counterfactual and non-counterfactual (i.e., original samples) for the two classes of Satisfaction and Dissatisfaction.

3.6.2. USER SATISFACTION ESTIMATION RESULTS

Table 3.6 shows the results of user satisfaction estimation using BERT and ASAP as the state-of-the-art models [75, 194], as well as two LLMs, `Zephyr` and `MistralIF`.

Data Partition	MultiWOZ	SGD
Satisfaction	64.6	86.2
Dissatisfaction	47.5	14.5
Overall	63.8	80.3

Table 3.4: Counter Satisfaction Status (CSS). CSS demonstrates the success rate of LLMs in generating counterfactual system utterances.

	MultiWOZ		SGD	
Label	Main	CF	Main	CF
#Satisfaction	811	19	848	11
#Dissatisfaction	40	524	76	731
#Total	851	543	924	742

Table 3.5: Statistics of original test samples (Main) and generated counterfactual samples (CF).

BERT and ASAP models are fine-tuned using the training samples indicated in Table 3.2. The two LLMs, however, are used in a few-shot manner as described in Section 3.4.2. We evaluate these models using different test sets. The Main group of results (at the top of Table 3.6) refers to the original test set from [172]; CF refers to the counterfactual version of Main, which is generated as described in Section 3.4.1; and Mix is the aggregation over both Main and CF.

As the table suggests, while on the original data (Main), which is highly imbalanced across *satisfaction* and *dissatisfaction* labels, BERT and ASAP outperform the two LLMs, in the rest of the test sets (CF, Mix), it is the LLMs that achieve higher performance than BERT and ASAP by a large margin. Moreover, while we can see a drastic drop in the performance of BERT and ASAP on CF in comparison to their performance on the Main set, the performance of LLMs on the two sets of Main and CF is comparable. These results show the robustness of few-shot in-context learning for user satisfaction estimation under different distributions of labels in the test data. In addition, we can see from the results on the CF test data that while increasing the ratio of up-sampling dissatisfaction training samples from 10x to 20x increases the performance of the BERT and ASAP estimators on the MultiWOZ dataset, this way of augmenting training samples does not have the same effect on the SGD test set. This may indicate the lack of proper training data and the necessity for augmenting the training data for fine-tuning user satisfaction estimators. Furthermore, it highlights the need for more sophisticated data augmentation approaches rather than simply up-sampling the data. It is noteworthy that we also conducted our experiments using under-sampling of the satisfactory class; however, the results corresponding to this approach are not included since it led to a weak performance.

Robustness results. The Main and CF test collections (Table 3.5) are the two

			MultiWoZ				SGD			
Test Data	**Model**	**Setup**	Acc	P	R	F1	Acc	P	R	F1
Main	BERT	w/o up-sampling	**95.30**	47.65	50.00	48.80	91.34	45.87	49.76	47.74
	BERT	up-sampling x10	93.88	61.46	57.58	59.02	83.55	57.85	62.89	59.17
	BERT	up-sampling x20	92.36	54.99	54.40	54.67	89.72	58.39	54.27	55.23
	ASAP	w/o up-sampling	94.95	**71.87**	**72.39**	**72.13**	**92.10**	**73.77**	63.35	66.69
	ASAP	up-sampling x10	93.30	65.23	69.15	66.91	86.15	64.41	75.68	67.49
	ASAP	up-sampling x20	90.95	61.31	70.30	64.10	86.58	65.05	**76.52**	**68.26**
	Zephyr	Few-shot	73.80	51.56	56.54	48.23	84.63	52.36	52.70	52.49
	MistralIF	Few-shot	80.14	51.92	56.31	50.62	87.01	53.98	53.39	53.63
CF	BERT	w/o up-sampling	3.50	1.75	50.00	3.38	2.83	50.75	50.68	2.83
	BERT	up-sampling x10	8.66	51.84	52.67	8.63	21.43	50.93	60.12	18.66
	BERT	up-sampling x20	12.34	51.92	54.58	12.09	4.18	50.76	51.37	4.16
	ASAP	w/o up-sampling	4.24	30.03	25.02	4.23	4.99	47.30	42.82	4.92
	ASAP	up-sampling x10	6.63	38.96	31.33	6.57	16.44	49.36	44.16	14.70
	ASAP	up-sampling x20	9.94	41.17	25.44	9.50	12.67	48.34	37.77	11.64
	Zephyr	Few-shot	**88.95**	**61.58**	**91.74**	**65.72**	**83.69**	**54.17**	**91.72**	**53.18**
	MistralIF	Few-shot	82.32	57.85	88.30	58.60	73.72	52.67	86.66	47.37
Mixed	BERT	w/o up-sampling	59.54	29.77	50.00	37.32	51.92	61.59	50.39	35.27
	BERT	up-sampling x10	60.69	62.67	51.96	43.04	55.88	58.62	54.85	49.87
	BERT	up-sampling x20	61.19	62.44	52.89	45.56	51.62	51.26	50.17	37.03
	ASAP	w/o up-sampling	59.61	55.41	51.00	42.21	53.30	61.48	51.88	39.84
	ASAP	up-sampling x10	60.83	59.69	52.87	46.47	53.42	54.70	52.31	45.93
	ASAP	up-sampling x20	59.40	54.85	51.73	45.81	53.66	55.21	52.55	46.16
	Zephyr	Few-shot	79.70	79.47	**80.57**	79.46	84.21	**84.88**	**83.99**	**84.06**
	MistralIF	Few-shot	80.99	**80.24**	80.54	**80.37**	81.09	83.26	80.69	80.62

Table 3.6: User satisfaction estimation results on MultiWOZ and SGD using binary satisfaction and dissatisfaction labels. Metrics are based on macro averaging. Main is the original test data in the benchmarks, CF refers to the counterfactual version of the original test data (with flipped user satisfaction labels), and Mix is the combination of Main and CF. Few-shot refers to the few-shot in-context learning with LLMs. For each dataset (Main, CF, Mixed) the best and second best results are pointed out in **bold** and underline, respectively.

extremes in case of imbalance in the test data for the number of satisfaction and dissatisfaction test samples. To better explore the robustness of models with varying numbers of test samples from the two classes of *Satisfaction* and *Dissatisfaction*, we evaluate the models using different proportions of these classes. To this aim, we start with the Main test set with an approximate 95:5 ratio for satisfaction:dissatisfaction labels. We then increase the number of dissatisfaction labels in the Main condition using the dissatisfaction dialogue samples from the CF condition. We evaluate models while increasing the dissatisfaction fraction in steps of 5%. Figure 3.4 depicts the performance of all models on the MultiWOZ and SGD benchmarks. We see that the performance of the fine-tuned state-of-the-art models (BERT and ASAP)

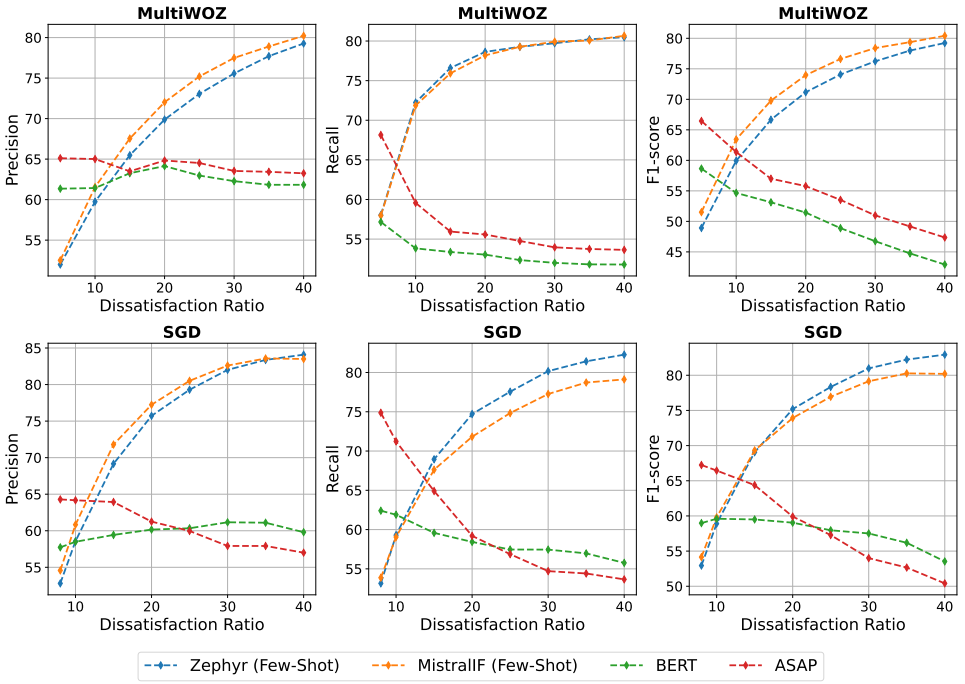

Figure 3.4: Performance of USE models with a varying degree of imbalance in the test set for the MultiWOZ and SGD benchmarks. The dissatisfaction ratio is the proportion of samples with *dissatisfaction* labels in the test collection.

drastically drops when more *Dissatisfaction* samples are included in the evaluation. Moreover, Figure 3.5 shows the sensitivity (recall) for only the *Dissatisfaction* class. As we can see, few-shot in-context learning with LLMs provides an increased ability to identify user dissatisfaction in the dialogues, which is a crucial factor in the deployment of dialogue systems. This is particularly important as we can see the higher performance of fine-tuned state-of-the-art models (BERT and ASAP) in comparison to LLMs on the original test set (Main in Table 3.6), which includes about 5% dissatisfaction samples. However, the sensitivity of these fine-tuned state-of-the-art models (BERT and ASAP) for the identification of user dissatisfaction is either lower than LLMs (BERT versus LLMs on MultiWOZ in Figure 3.5) or becomes comparable with them with a slight increase in the number of *Dissatisfaction* samples, e.g., change in results from 5% to 10% dissatisfaction ratio in Table 3.5.

Shared-context results. The counterfactual dialogue samples in the CF test set differ from the corresponding original samples in the Main test set in terms of the last system response (see Figure 3.2). To measure the success rate of estimators in predicting the user satisfaction label for both a dialogue and its corresponding counterfactual sample, i.e., two samples with the same context (dialogue history), we use the Jaccard similarity index (JSI) $\frac{|M \cap C|}{|M \cup C|}$, where M and C are the correctly

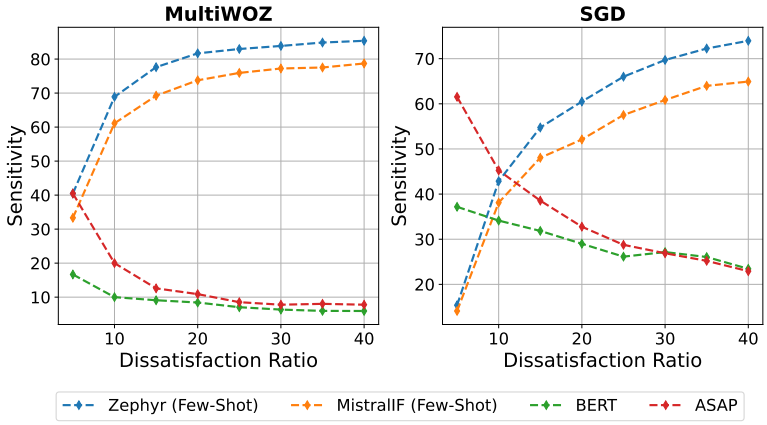

Figure 3.5: Sensitivity of the models in identification of user dissatisfaction on various proportions of *dissatisfaction* test samples.

predicted samples of the Main and CF test collections respectively. Table 3.7 shows the JSI for different user satisfaction estimators. The best performing BERT and ASAP setups from Table 3.6 are selected for this purpose. As the table shows, BERT and ASAP have a very low JSI in comparison to the LLM-based satisfaction estimators which is in line with the result of these models on the Main and CF test sets in Table 3.6. Furthermore, we can see that on both the MultiWOZ and SGD test sets, `Zephyr` has a higher JSI than `MistralIF`, even though `MistralIF` outperforms `Zephyr` on the Main test set (top-rows in Table 3.6).

Model	MultiWOZ	SGD
BERT	0.0419	0.1551
ASAP	0.0166	0.0512
Zephyr	0.7332	0.7538
MistralIF	0.6282	0.6638

Table 3.7: Shared-context results (Jaccard Similarity Index) of user satisfaction estimation.

3.7. CONCLUSION

We have studied the task of user satisfaction estimation and specifically focused on the robustness of estimators for TOD systems. We augment two previously introduced benchmarks using satisfaction-focused counterfactual utterance generation and conduct human evaluation on the generated dialogues. Using our augmented test collections, we show that there is a discrepancy between the performance of estimators on the original test sets and the test sets with a higher ratio of *dissatisfaction* dialogue samples.

Our experiments highlight an important missing aspect in previous studies: the robustness of satisfaction estimators for the identification of user dissatisfaction. Moreover, this chapter sheds light on the need for further research on data augmentation for training user satisfaction estimators. We hypothesize that training models with more balanced data is beneficial for the robustness of these models. In this chapter, we also unlock the power of LLMs in generating quality counterfactual dialogue samples which seems to be a promising direction for augmenting the training set of user satisfaction estimators. In future work, we plan to leverage LLMs for such satisfaction-oriented data augmentation in TOD systems. Furthermore, in this chapter, we only work on turn-level satisfaction estimation and leave the dialogue-level setting for future work as generating dialogue-level counterfactual data requires more sophisticated methods. Finally, we have explored user satisfaction estimation only in task-oriented dialogue systems. User satisfaction estimation has also been studied for other tasks including conversation recommender systems [163, 172]. Also, we plan to study counterfactual utterance generation for a more broad application of USE in dialogue systems.

LIMITATIONS

While we employ proprietary model `GPT-4` for the generation of counterfactual samples, we also point out the limitation in this approach in the sense that it still requires to leverage of a proprietary LLM. Here, we should note that we use `GPT-4` to create counterfactual data samples in order to enhance the existing benchmarks. This is a one-off usage of proprietary models that enables future research on the evaluation of user satisfaction estimation for task-oriented dialogue systems.

In addition, it should be noted that our current research is exclusively on datasets in English. Therefore, we highlight the necessity of extending our experiments to include datasets in languages other than English. This expansion is of importance to ensure the applicability of our findings across a broader linguistic spectrum.

3.A. APPENDIX

3.A.1. COUNTERFACTUAL RESPONSE GENERATION PROMPT

Figure 3.6 shows the prompt used to generate a counterfactual system utterance for a new input dialogue, using few-shot in-context learning with `GPT-4`.

3.A.2. FULL-DIALOGUE COUNTERFACTUAL EXAMPLES

Figure 3.7 illustrates two dialogue samples with both their original last system utterance and the counterfactual last system utterance.

```
You are a counterfactual utterance generator which receives a dialogue and
generates a counterfactual utterance for the 'last' SYSTEM utterance in the
dialogue.

Here is the definition of counterfactual SYSTEM utterance:  If in the
last SYSTEM utterance, SYSTEM fails to fulfill or understand the USER request in
any way, in a counterfactual SYSTEM utterance, SYSTEM understand and fulfills the
USER request.
On the contrary, if in the last SYSTEM utterance, SYSTEM understand and fulfills
the USER request, in a counterfactual SYSTEM utterance, SYSTEM fails to understand
and fulfill the USER request.

Here are two Example dialogues:

Example 1:
USER: Is it true Cambridge has the best attractions?  We are looking for something
unusual to do in the centre.
SYSTEM: There is teh museum of archeology and anthropology in the centre that is
free of charge
USER: That's perfect.  Thanks so much for your help.
SYSTEM: You're welcome.  Is there anything else I can assist you with?
USER: Yes can you help me find a place to eat too?
SYSTEM: I'd be happy to help with your request, what area and what type of cuisine
are you looking for?
USER: I would like chinese.
SYSTEM: I'm afraid our restaurant system is currently down.  Can I help you with
something else at this time?

Counterfactual SYSTEM last utterance for Example 1:
SYSTEM: Sure, for Chinese cuisine, I recommend the Golden Dragon located in the
city center.  It's known for its authentic flavors and has excellent reviews.
Would you like me to book a table for you?

Example 2:
USER: I need to book a train from bishops stortford to Cambridge on Saturday
arriving in Cambridge before 12:30.
SYSTEM: I have TR4594, leaving at 11:29 and arriving in 12:07.  Is that OK?
USER: I actually need to leave after 20:30 on Friday.  So whatever is closest to
that time will be fine.
SYSTEM: Okay.  The TR4549 leaves at 21:29.  Will that suit?
USER: Is there bike parking at the train station?
SYSTEM: Bike parking is available at the station.  Are you still interested in a
reservation?

Counterfactual SYSTEM last utterance for Example 2:
SYSTEM: Sorry, the information regarding the Bike parking is not available.
Would you like to look for alternative transportation options?

Now, generate a counterfactual utterance for the 'last' SYSTEM utterance in the
following dialogue:

{Input Dialogue}
```

Figure 3.6: The input prompt used for generating counterfactual utterance for the last system utterance of a given dialogue.

Dissatisfaction → Satisfaction

USER: I need to find a guesthouse with a 3 star rating
SYSTEM: We have four such guesthouses. Do you have a preferred location?
USER: No, I don't. I want one that includes free wifi.
SYSTEM: Bridge Gue House, Hamilton Lodge, and Hobsons House are all available if you'd like one of those?
USER: Do you know whether they provide daily housekeeping service?
SYSTEM: The Hobsons House offers housekeeping service daily. Can I offer any more assistance?
USER: What is the fee for staying with pets?
SYSTEM: The Hobsons House does not allow pets. May I help you with anything else?
USER: Does the Hobsons House offer WiFi for guests?
SYSTEM: Free Wifi is available at the Hobsons House. What other specifications are you looking for?
USER: Please book the Hamilton Lodge, for 6 people.
SYSTEM: I'd be happy to. How many nights would you like to stay? And I'll need to know the day you would like to start your stay.
USER: I'll be staying 2 nights starting on wednesday. Oh, and there are 6 of us.

Original System Response **R** [Dissatisfaction]:
SYSTEM: I'm sorry, my reservation attempt was unsuccessful. Will another day or length of stay work?

Counterfactual System Response $\widehat{\mathbf{R}}$ [Satisfaction]:
SYSTEM: Great, I have successfully booked the Hamilton Lodge for 6 people for 2 nights starting on Wednesday. Your reservation is confirmed. Is there anything else you need assistance with?

Satisfaction → Dissatisfaction

USER: Would you be able to help me find a room that has free wifi?
SYSTEM: I certainly can. Do you have additional preferences?
USER: I'm looking for a guesthouse, and I misspoke earlier. I actually don't care about the internet, but I do need free parking.
SYSTEM: I have 21 guest houses, can you tell me what area you would like to be in?
USER: The area doesn't matter.
SYSTEM: Ok. The acorn guest house is in the north part of town in the moderate price range. Would that work?
USER: Does the Acorn provide any kind of kitchen equipment that guests can use?
SYSTEM: There is no kitchen available here. Can I still book a reservation for you?
USER: Yes. I would like to book that for 4 people for 4 nights staring on Wednesday.
SYSTEM: Sorry, it was unavailable at that time. Perhaps you might want another day or a shorter stay?
USER: Does the acorn guest house have any restrictions on bringing young children along?

Original System Response **R** [Satisfaction]:
SYSTEM: There are no restrictions. Children are welcome to stay there. Any other questions, or should I continue to book it?

Counterfactual System Response $\widehat{\mathbf{R}}$ [Dissatisfaction]:
SYSTEM: I'm not sure about their policy on young children. Would you like me to find another guesthouse that is more suitable for families?

Figure 3.7: Examples of generated counterfactual system utterances. Satisfaction to Dissatisfaction (top) and vice versa (bottom).