MultiWOZ 2.3: A multi-domain task-oriented dataset enhanced with annotation corrections and co-reference annotation

Ting Han[1,2*] **Ximing Liu**[2*] **Ryuichi Takanobu**[3] **Yixin Lian**[2]
Chongxuan Huang[2] **Wei Peng**[2†] **Minlie Huang**[3†]
[1]University of Illinois at Chicago
[2]Artificial Intelligence Application Research Center (AARC), Huawei Technologies
[3]Tsinghua University
than24@uic.edu, liuximing1@huawei.com, gxly19@mails.tsinghua.edu.cn
{huang.chongxuan, lianyixin1, peng.wei1}@huawei.com
aihuang@mail.tsinghua.edu.cn

Abstract

Task-oriented dialogue systems have made unprecedented progress with multiple state-of-the-art (SOTA) models underpinned by a number of publicly available MultiWOZ datasets. Dialogue state annotations are error-prone, leading to sub-optimal performance. Various efforts have been put in rectifying the annotation errors presented in the original MultiWOZ dataset. In this paper, we introduce MultiWOZ 2.3, in which we differentiate incorrect annotations in dialogue acts from dialogue states, identifying a lack of co-reference when publishing the updated dataset. To ensure consistency between dialogue acts and dialogue states, we implement co-reference features and unify annotations of dialogue acts and dialogue states. We update the state of the art performance of natural language understanding and dialog state tracking on MultiWOZ 2.3, where the results show significant improvements than on previous versions of MultiWOZ datasets (2.0-2.2).

1 Introduction

Task-oriented dialogue systems have made unprecedented progress with multiple state-of-the-art (SOTA) models underpinned by a number of publicly available datasets (CrossWOZ (**?**), DSTC (Henderson et al., 2014; Williams et al., 2013), WOZ (Wen et al., 2017), SGD (**?**), MultiWOZ (**?**)).

As the first publicly released dataset, MultiWOZ hosts more than 10K dialogues across eight different domains covering “Train”, “Taxi”, “Hotel”, “Restaurant”, “Attraction”, “Hospital”, “Bus” and “Police”. MultiWOZ has been widely adopted by researchers in dialogue policy (**?**Zhao et al., 2019), dialogue generation (Chen et al., 2019) and dialogue state tracking (**???**Lee et al., 2019a; Wu et al., 2019) as it provides a means for modeling the changing states of dialogue goals in multi-domain interactions.

Dialogue state annotations are error-prone, leading to sub-optimal performance; for example, the SOTA joint accuracy for dialogue state tracking (DST) is still below 60%[1]. MultiWOZ 2.1 (**?**) was released to rectify annotation errors presented in the original MultiWOZ dataset. MultiWOZ 2.1 introduced additional features such as slot descriptions and dialogue act annotations for both systems and users via ConvLab (Lee et al., 2019b). Further efforts have been put into MultiWOZ 2.2 (**?**) to improve annotation quality. This schema-based dataset contains annotations allowing for directly retrieving slot values from a given dialogue context (**???**). Despite achieving a noticeable annotation quality uplift compared to that for the original MultiWOZ, there is still space to improve. The focus of the corrections is on dialogue state annotations leaving the problematic dialogue act annotations untouched. Furthermore, the critical co-reference and ellipsis feature prevalent in the human utterance is not in presence.

To address the limitations above, we introduce an updated version, MultiWOZ 2.3. Our contributions are as follow:

- We differentiate incorrect annotations in dialogue acts from dialogue states, and unify annotations of dialogue acts and dialogue states to ensure their consistency when publishing the updated dataset, MultiWOZ 2.3.
- We introduce co-reference features to enhance the performances of dialogue systems in the new version.

*Both authors contributed equally to the work. The work was conducted when Ting Han was interning at Huawei AARC.

†Corresponding author.

[1]https://github.com/budzianowski/multiwoz. Marked date: 10/12/2020

- We re-benchmark a few SOTA models for dialogue state tracking (DST) and natural language understanding (NLU) tasks and provide a fair comparison using the updated dataset.

2 Annotation Corrections

The inconsistent annotations in the MultiWOZ dataset were caused by disparate interpretations from involved annotators during a crowdsourcing process. These errors can occur even when annotators attempt to apply unified rules. After analyzing annotation errors in both dialogue acts and dialogue states, we perform the following two data corrections.

2.1 Dialogue Act Corrections

The annotations for user dialogue acts were originally introduced by Lee et al. (2019b). Following the pipeline provided in ConvLab, **?** re-annotated dialogue acts for both systems and users in MultiWOZ 2.1. We broadly categorize the incorrect annotations into three types (Figure 1) based on our observations:

- **Under-annotated**: Annotation errors under this category are due to insufficient annotation even when exact information is available in the given dialogue utterances. The missing annotations should be added to the corresponding slots.
- **Over-annotated**: Sometimes, incorrect annotations are put down even when no corresponding information can be identified in the utterances. The over-annotated values should be removed to avoid confusion.
- **Wrongly-annotated**: This category refers to slots with incorrect values (or span information) and should be fixed.

We apply two rules to sequentially correct "dialog_act" annotations: a) we use customized filters to select credible predictions generated from a MultiWOZ 2.1 pre-trained BERTNLU model (**?**) and merge them with original "dialog_act" annotations; b) we use assorted regular expressions to further clean "dialog_act" annotations from the previous step.

To fairly evaluate the quality of modified annotations, we sampled 100 dialogues from the test set and manually re-annotated the dialogue acts. Table 1 exhibits the ratios of "dialog_act" annotations of different datasets in terms of slot level and turn level using the manually-annotated 100 dialogues as golden annotations.

Version	**Rule**	**Slot Level**	**Turn Level**
2.1/2.2	Strict	77.59%	68.83%
	Relax*	82.94%	77.19%
2.3	Strict	84.12%	76.09%
	Relax*	90.74%	86.83%

Table 1: A comparison of annotation correctness ratios of "dialog_act" for MultiWOZ 2.1/2.2 and 2.3. The "Relax" rule indicates that the values of insignificant slots like "general-xxx" and "none" are removed.

We added 24,405 slots and removed 4,061 slots in the "dialog_act" annotations. Roughly 16,800 slots are modified according to our estimation. Also note that in Figure 1, boundaries for the three types are not strictly drawn. *PMUL2596.json* under wrongly-annotated type can also be treated as an under-annotated error when slot *Taxi-Inform.Dest* is missing.

Adding and removing operations for "dialog_act" annotations cause mismatches in paired span indices. When aligning span information with the modified dialogue acts, we note that original span information also contains incorrect annotations, such as abnormal span with ending index ahead of the starting index, incorrect span, and drifted span. The errors are all corrected, along with those for dialogue acts.

2.2 Dialogue State Corrections

The fixed "dialog_act" and the "span_info" annotations are propagated into the dialogue state annotations, i.e., " metadata" because we need to maintain the consistency among them.

Since the repairing for dialogue states is based on cleaned dialogue acts, we use the following rules to guide updating dialogue state annotations (Figure 2):

- **Slot Value Normalization**: Multiple slots values exist in MultiWOZ 2.2 due to a mismatch between given utterances and ontology, for example, "16:00" and "4 PM". This potentially leads to incomplete matching, as the values are not normalized. We follow the way MultiWOZ 2.1 in normalizing slot values to base on utterances to this end.

- **Consistent Tracking Strategy**: The inconsistent tracking strategy (Figure 3) was initially discussed (but not solved) in MulitWOZ 2.2. We track the user's requirements from slot values informed by the user, recommended by the system, and implicitly agreed by the user. We apply two sub-rules to resolve the implicit agreements: a) if an informing action is from users to systems, the informed values are propagated to the next turn of dialogue states; b) if an informing/recommending action is from systems to users, the informed or recommended values are propagated to the next turn of dialogues states if and only if one item is included. Multiple items are not considered to be valid in the implicit agreement settings.

Fixing Type	Count	Ratio
No Change	2,476,666	98.68%
Value Filled	20,639	0.82%
Value Changed	11,649	0.46%
Value Removed	221	0.01%
Value dontcare	563	0.02%

Table 2: Percentage of slots' values changed in MultiWOZ 2.3 and MultiWOZ2.1, respectively, for "metadata" annotations. "Value Filled" stands for a value-filled from null, "none" or "not mentioned". "Value Removed" means a slot value is changed to "not mentioned" or null. "Value dontcare" stands for slot values filled with "dontcare".

Table 2 shows statistics on the type of corrections we have made on the "metadata" annotations. Note that "dontcare" value is singled out during repairing since it is a significant factor (Table 6) on slot gate classifications in the TRADE model (Wu et al., 2019).

3 Enrich Dataset with Co-referencing

MultiWOZ contains a considerable amount of co-reference and ellipsis. As shown in Tabel 3, co-referencing frequently occurs in the cross-domain dialogues, especially when aligning the value of "Name" slot from a hotel (or restaurant) domain with those of "Departure/Destination" slots for taxi/train domains. The lack of co-reference annotations leads to poor performances presented in existing DST models.

A number of task-oriented dialogue models leveraged datasets enriched with co-referencing features to achieve SOTA results (**?**). By including co-reference in CamRest676 (Wen et al., 2017), GECOR (**?**) showed significant performance improvement compared to the baseline models. Through restoring incomplete utterances by annotating the dataset with co-reference labels, **?** boosted response quality of dialogue systems. Su et al. (2019) re-wrote utterances to cover co-referred and omitted information to realize notable success on their proposed model.

In MultiWOZ 2.1, the distributions of co-referencing among different slots are presented in Table 4. In total, 26.52% dialogues are annotated with co-reference in the dataset, indicating the importance of co-referencing annotation.

3.1 Annotation for Co-reference

The "coreference" annotations are applied to all "dialog_act" slots having co-referencing relationships with other slots. The annotation takes a "Domain-Intent" format, including five parts: slot name, slot value in the current turn, referred value, referred turn id, and spans of referred value in the referred turn. Figure 4 depicts an example of "coreference" annotation.

PMUL4852.json


```
▼ 10:
    text: "That sounds wonderful ! Is it in the same area as the hotel ?"
    metadata: {}
  ▸ dialog_act:
  ▸ span_info: [] 1 item
  ▾ coreference:
    ▾ Hotel-Inform: [] 1 item
      ▾ 0: [] 5 items
          0: "Area"
          1: "same area"
          2: "center"
          3: 4
          4: "12-12"
    turn_id: 10
```


Figure 4: Example of a co-referencing annotation. If the current turn involves more than one co-referencing relationships, all annotations will be gathered under the "coreference" key. The number "10" at the top left corner indicates the "turn_id" of dialogue *PMUL4852.json*.

We apply co-referencing annotations to problematic slots when necessary, for example, "Area/Price/People/Day/Depart/Dest/Arrive". The co-referencing annotations are added sequentially:

- We use first regular expressions to locate co-reference slots;

- Based on the current dialogue states, we trace

Sample dialogue ID	Utterances
PMUL1815.json	I'm traveling to Cambridge from london liverpool street arriving by 11:45 the day *(saturday)* of my hotel booking.
PMUL2049.json	Thank you, can you also help me find a restaurant that is in the same area *(centre)* as the Parkside pools?
PMUL2512.json	Thanks! I'm going to hanging out at the college *(christ college)* late tonight, could you get me a taxi back to the hotel *(the express by holiday inn cambridge)* at 2:45?

Table 3: Examples of co-reference annotations. Co-reference values are added to the original utterances and marked as light orange italic inside the brackets.

back to the history utterances where the co-referred slots are first encountered;

- we use corresponding dialogue acts with paired span information to retrieve co-referred values.

In total, we added 3,394 co-referencing annotations for "dialog_act".

Slot	Count	Ratio
Taxi.Depart	844	24.82%
H/R/A.Area	786	23.12%
Taxi.Dest	706	20.76%
H/R/A/T.Day	409	12.03%
H/R.Price	354	10.41%
H/R/T.People	201	5.91%
Taxi.Arrive	92	2.71%

Table 4: Statistics of co-reference annotations. H/R/A/T represent "Hotel", "Restaurant", "Attraction" and "Train", respectively.

Table 4 shows the statistics of the amount of "coreference" annotations for each slot type. We can see the most common co-referencing relationship is from "Taxi-Dest/Depart" and "xxx-Area", followed by "Day", "Price", "People" and "Arrive".

4 Benchmarks and Experimental Results

The updated dataset is evaluated for a natural language understanding task and a DST task. Experiment results are produced to re-benchmark a few SOTA models.

4.1 Dialogue Actions with Natural Language Understanding Benchmarks

BERTNLU (**?**) is introduced for dialogue natural language understanding. It tops extra two multilayer perceptron (MLP) layers on BERT (Devlin et al., 2019) for slot recognition and intent classification (**?**), respectively. In practice, BERTNLU achieves better performance on classification and tagging tasks by including historical context and finetuning all parameters. We implement BERTNLU with inputs of current utterance plus the previous three history turns and finetune it based on the dialogue act annotations. The model's performance is evaluated by calculating F1 scores for intents, slots, and for both. Additionally, we use utterance accuracy as another metric to assess the model's effectiveness in understanding what the user expresses in an utterance. We score each utterance either 0 or 1 according to whether the predictions of all the slots, intents, or both in an utterance match the correct labels. The utterance accuracy is characterized as the average of this score across all utterances. Table 5 shows the performance of BERTNLU on different datasets (including dialogue utterances from both user and system sides) based on the above evaluation metrics.

4.2 Dialogue State Tracking Benchmarks

Multiple neural network-based models have been proposed to improve joint goal accuracy of dialogue state tracking tasks. Existing belief state trackers could be roughly divided into two classes: span-based and candidate-based. The former approach (**??**Lee et al., 2019a) directly extracts slot values from dialogue history, while the latter approach (Wu et al., 2019) is to perform classification on candidate values, assuming all candidate values are included in the predefined ontology. To evaluate our updated dataset for DST task, we run experiments on TRADE (Wu et al., 2019) and SUMBT (Lee et al., 2019a).

SUMBT uses a multi-head attention mechanism to capture relations between domain-slot types and slot values presented in the utterances. The attend-

Dataset	F1(Slot/Intent/Both)	Utter. Acc.(Slot/Intent/Both)
MultiWOZ 2.1	81.18/88.34/83.77	81.89/86.23/71.68
MultiWOZ 2.2	80.61/88.34/83.41	81.94/86.41/71.85
MultiWOZ 2.3	**89.03/90.73/89.65**	**87.33/88.56/78.33**

Table 5: Performance of BERTNLU on different datasets based on F1 score and utterance accuracy for slots, intents and both, respectively. Utterance accuracy is defined as the average accuracy of predicting all the slots, intents or both in an utterance correctly.

Dataset	Pointer(P/R/F1)	Dontcare(P/R/F1)	None(P/R/F1)
MultiWOZ 2.1	94.97/93.75/94.35	58.73/32.51/41.85	98.25/98.82/98.53
MultiWOZ 2.2	94.22/94.42/94.32	60.21/34.60/43.91	98.42/98.64/98.53
MultiWOZ 2.3	**96.41/96.15/96.28**	**67.80/41.62/51.58**	**98.79/99.11/98.95**

Table 6: Classification on slot gate for TRADE using different datasets. “Pointer”, “dontcare” and “none” are three different slot gate classes. Precision, recall, and F1-score are used as metrics to evaluate among all datasets.

ed context words are collected as slot values for corresponding slots. TRADE uses a pointer to differentiate, for a particular domain-slot, whether the slot value is from the given utterance or the predefined vocabulary. Both models perform predictions slot by slot and treat all slots equally.

Following the convention in dialogue state tracking task, joint goal accuracy is used to evaluate the models’ performances for different datasets. The models also experiment with co-referencing enhanced datasets. Table 7 summarizes the joint goal accuracy of the two models using different datasets.

Dataset	SUMBT	TRADE
MultiWOZ 2.0	46.6%♦	48.6%♦
MultiWOZ 2.1	49.2%	45.6%
MultiWOZ 2.2	49.7%	46.6%
MultiWOZ 2.3	**52.9%**	**49.2%**
MultiWOZ 2.3-cof	**54.6%**	**49.9%**

Table 7: Joint goal accuracy of SUMBT and TRADE over different versions of dataset. MultiWOZ 2.3-cof refers to the dataset with co-reference applied. ♦ means the accuracy scores are adopted from the published papers.

4.3 Experimental Analysis

As shown in Tables 5 and 7, substantial performance increases are achieved with the enhanced datasets compared to the previous datasets. BERTNLU trained using our dataset outperforms others with a margin of 5% improvement on both metric of F1-score and utterance accuracy. In the task of DST, models trained using our datasets also show superiority to those trained with the previous version MultiWOZ. By applying co-referencing features to dialogue state tracking, the joint goal accuracy is improved to approximately 55% using SUMBT.

5 Discussion

Note that SUMBT initially focused on MultiWOZ 2.0. Fixing dialogue states leads to enhanced data quality in MultiWOZ 2.1. This study designs a new pre-process script to correct the identified errors in MultiWOZ 2.1 further. The joint goal accuracy can reach 54.54%, and 56.09% for co-reference augmented utterances using SUMBT. Since multiple slot values are allowed for MultiWOZ 2.2, it is not practical to identify errors in the dialogue states. We do not base this study on MultiWOZ 2.2 at this stage. Figure 5 shows pairwise comparisons between two datasets on the benchmarked scores. Our dataset (MultiWOZ 2.3) tops all the scores compared to previously updated datasets in all MultiWOZ specified slots. Details of slot accuracies are presented in Figures 6 and 7) at Appendix B. As shown in Figure 6, our dataset achieves the best performance for 17 out of all 30 slots. The performance is further enhanced with the co-reference version (in Figure 7).

Table 6 shows precision, recall, and F1-score of slot gate classifications in the TRADE model across different datasets. For the three different classes, our dataset achieves top performances. As a result of the carefully designed error correction (Table 8 in Appendix A), our dataset outperforms others by at least 9% in all metrics for the “dontcare” gate.

Based on the contexts presented in utterances,

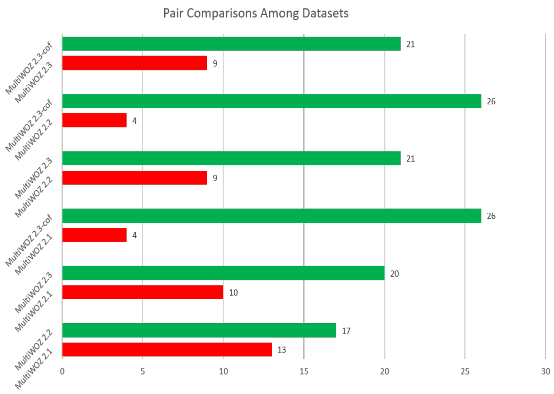

Figure 5: Pairwise comparison between two datasets in terms of the number of higher accuracy slots. In total, there are 30 valid slots in the DST task. The number on top of each bar indicates the number of winning slots in comparison.

we have fixed the dialogue acts and removed the inconsistency between dialogue acts and states. Span indices in the dialogue acts are further fixed with co-reference information introduced. By closely aligning the annotations to corresponding utterances mentioned above, we remove the inconsistency introduced by annotating a Wizard-Of-Oz dataset.

6 Conclusion

MultiWOZ datasets (2.0-2.2) are widely used in dialogue state tracking and other dialogue related subtasks. Mainly based on MultiWOZ 2.1, we publish a refined version, named MultiWOZ 2.3. After correcting annotations for dialogue acts and dialogue states, we introduce co-reference annotations, which supports future research to consider discourse analysis in building task-oriented dialog systems. We re-benchmark the refined dataset using some competitive models. The experimental results show significant improvements for the associated scores, verifying the utility of this dataset. We hope to attract more alike research works to improve the quality of MultiWOZ datasets further.

References

Wenhu Chen, Jianshu Chen, Pengda Qin, Xifeng Yan, and William Yang Wang. 2019. Semantically conditioned dialog response generation via hierarchical disentangled self-attention. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3696–3709, Florence, Italy. Association for Computational Linguistics.

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.

Matthew Henderson, Blaise Thomson, and Jason D. Williams. 2014. The second dialog state tracking challenge. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 263–272, Philadelphia, PA, U.S.A. Association for Computational Linguistics.

Hwaran Lee, Jinsik Lee, and Tae-Yoon Kim. 2019a. SUMBT: Slot-utterance matching for universal and scalable belief tracking. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5478–5483, Florence, Italy. Association for Computational Linguistics.

Sungjin Lee, Qi Zhu, Ryuichi Takanobu, Zheng Zhang, Yaoqin Zhang, Xiang Li, Jinchao Li, Baolin Peng, Xiujun Li, Minlie Huang, and Jianfeng Gao. 2019b. ConvLab: Multi-domain end-to-end dialog system platform. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 64–69, Florence, Italy. Association for Computational Linguistics.

Hui Su, Xiaoyu Shen, Rongzhi Zhang, Fei Sun, Pengwei Hu, Cheng Niu, and Jie Zhou. 2019. Improving multi-turn dialogue modelling with utterance ReWriter. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 22–31, Florence, Italy. Association for Computational Linguistics.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449, Valencia, Spain. Association for Computational Linguistics.

Jason Williams, Antoine Raux, Deepak Ramachandran, and Alan Black. 2013. The dialog state tracking challenge. In *Proceedings of the SIGDIAL 2013 Conference*, pages 404–413, Metz, France. Association for Computational Linguistics.

Chien-Sheng Wu, Andrea Madotto, Ehsan Hosseini-Asl, Caiming Xiong, Richard Socher, and Pascale Fung. 2019. Transferable multi-domain state generator for task-oriented dialogue systems. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 808–819, Florence, Italy. Association for Computational Linguistics.

Tiancheng Zhao, Kaige Xie, and Maxine Eskenazi. 2019. Rethinking action spaces for reinforcement learning in end-to-end dialog agents with latent variable models. 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 1208–1218, Minneapolis, Minnesota. Association for Computational Linguistics.

Error Type	Dialogue	Utterance	2.1 Dialog_act	2.3 Dialog_act
Under-annotated	SSNG0348.json	For 3 people starting on Wednesday and staying 2 nights .	Hotel-Inform.Stay: 2	Hotel-Inform.Stay: 2 Hotel-Inform.Day: Wednesday Hotel-Inform.Day: 3
	PMUL1170.json	Yes , one ticket please , can I also get the reference number ?	Train-Inform.People: 1	Train-Inform.People: one Train-Request.Ref: ?
	SNG01856.json	no, i just need to make sure it's cheap. oh, and i need parking	Hotel-Inform.Parking: yes	Hotel-Inform.Parking: yes Hotel-Inform.Price: cheap
Wrongly-annotated	PMUL2596.json	I will need to be picked up at the hotel by 4:45 to arrive at the college on tuesday .	Taxi-Inform.Leave: 04:45 Taxi-Inform.Depart: arbury lodge guesthouse Hotel-Inform.Day: tuesday	Taxi-Inform.Leave: 4:45 Taxi-Inform.Dest: the college Taxi-Inform.Depart: the hotel Hotel-Inform.Day: tuesday
	PMUL3296.json	Yeah , could you book me a room for 2 people for 4 nights starting Tuesday ?	Hotel-Inform.Stay: 2 Hotel-Inform.Day: Tuesday Hotel-Inform.People: 4	Hotel-Inform.Stay: 4 Hotel-Inform.Day: Tuesday Hotel-Inform.People: 2
	PMUL4899.json	How about funky fun house , they are located at 8 mercers row , mercers row industrial estate .	Attraction-Recommend.Name: funky fun house Attraction-Recommend.Addr: 8 mercers row Attraction-Recommend.Addr: mercers row industrial estate	Attraction-Recommend.Name: funky fun house Attraction-Recommend.Addr: 8 mercers row , mercers row industrial estate
Over-annotated	PMUL3250.json	No , I apologize there are no Australian restaurants in Cambridge . Would you like to try another type of cuisine ?	Restaurant-Request.Food: ? Restaurant-NoOffer.Food: Australian Restaurant-NoOffer.Area: Cambridge	Restaurant-Request.Food: ? Restaurant-NoOffer.Food: Australian
	MUL1118.json	If there is no hotel availability , I will accept a guesthouse. Is one available ?	Hotel-Inform.Type: guesthouse Hotel-Inform. Stars: 4	Hotel-Inform.Type: guesthouse
	MUL0666.json	Just please book for that room for 2 nights .	Hotel-Inform.Price: cheap Hotel-Inform.Stay: 2	Hotel-Inform.Stay: 2

Figure 1: Example of different error types of dialogue acts. The red color in the table highlights incorrect annotations and corresponding repaired results. Note that MultiWOZ 2.2 is excluded from the table because it added missing dialogue act annotations and the remainings are the same as MultiWOZ 2.1.

Dialogue ID	Utterance	MultiWOZ 2.1	MultiWOZ 2.3
MUL2602.json	*User:* Can you recommend me a nightclub where I can get jiggy with it? *Sys:* Well, I think the jiggiest nightclub in town is the Soul Tree Nightclub, right in centre city! Plus the entrance fee is only 4 pounds.	a-type=night club a-name=not mentioned a-area=not mentioned	a-type=nightclub a-name=not mentioned a-area=not mentioned
	User: That is perfect can I have the postcode please? *Sys:* Sure! The postcode is cb23qf	a-type=night club a-name=not mentioned a-area=not mentioned	a-type=nightclub a-name=soul tree nightclub a-area=not mentioned
MUL1455.json	*User:* I am also looking for a moderately priced chinese restaurant located in the north. *Sys:* Golden wok is in the moderate price range and in the north area would you like me to book it for you?	r-food=chinese r-pricerange=moderate r-name=not mentioned r-area=north	r-food=chinese r-pricerange=moderate r-name=not mentioned r-area=north
	User: Can I get the address and phone number please? *Sys:* Of course - the address is 191 Histon Road Chesterton cb43hl and the phone number is 01223350688.	r-food=chinese r-pricerange=moderate r-name=not mentioned r-area=north	r-food=chinese r-pricerange=moderate r-name=golden wok r-area=north

Figure 2: Example of updates on dialogue states. The red color in the figure highlights incorrect dialogue states and corresponding updated results. Note that MultiWOZ 2.2 is excluded from the figure because it is the same to MultiWOZ 2.1 in terms of inconsistent tracking. “a” and “r” used as slot names in the right two columns are abbreviations for “attraction” and “restaurant” respectively.

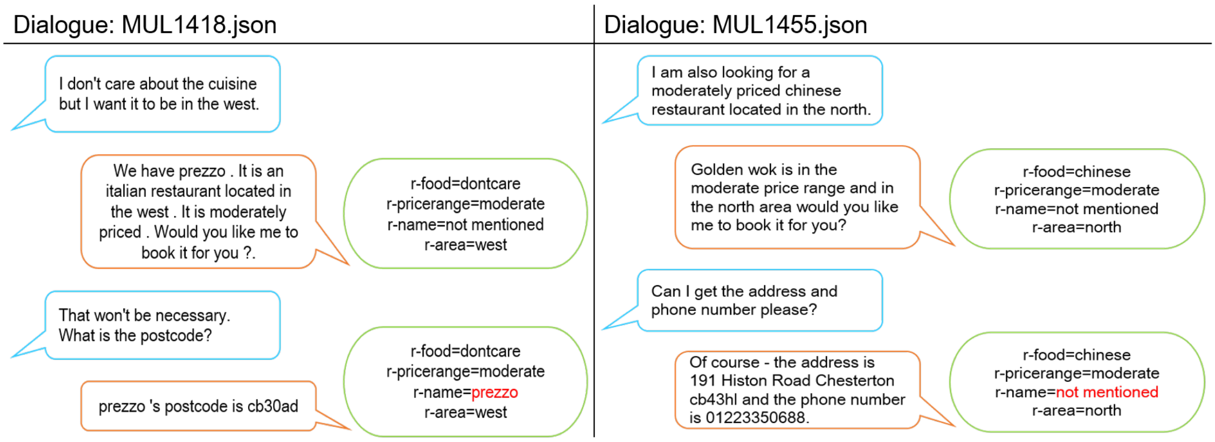

Figure 3: Examples of inconsistent tracking on dialogue states of two different dialogues in similar scenarios from MultiWOZ 2.1. In the left column, dialogue *MUL1418.json* updates slot *r-name* with "prezzo" recommended by the system. However, for dialogue *MUL1455.json* in the right column, the value of slot *r-name* is remained as "not mentioned" even though "golden wok" is recommended by the system. "r" in the light green rectangle is an abbreviation for "restaurant".

A Change Rules

Type	Content
Number	’zero’: ’0’, ’one’: ’1’, ’two’: ’2’, ’three’: ’3’, ’four’: ’4’, ’five’: ’5’, ’six’: ’6’, ’seven’: ’7’, ’eight’: ’8’, ’nine’: ’9’, ’ten’: ’10’, ’eleven’: ’11’, ’twelve’: ’12’
Pricerange	’high end’: ’expensive’, ’expensively’: ’expensive’, ’upscale’: ’expensive’, ’inexpensive’: ’cheap’, ’cheaply’: ’cheap’, ’cheaper’: ’cheap’, ’cheapest’: ’cheap’, ’moderately priced’: ’moderate’, ’moderately’: ’moderate’
dontcare	’do n’t have a preference’: ’dontcare’, ’do not have a preference’: ’dontcare’, ’no particular’: ’dontcare’, ’not particular’: ’dontcare’, ’do not care’: ’dontcare’, ’do n’t care’: ’dontcare’, ’any’: ’dontcare’, ’does not matter’: ’dontcare’, ’does n’t matter’: ’dontcare’, ’not really’: ’dontcare’, ’do nt care’: ’dontcare’, ’does n really matter’: ’dontcare’, ’do n’t really care’: ’dontcare’
Area	’center’: ’centre’, ’northern’: ’north’, ’northside’: ’north’, ’eastern’: ’east’ ’eastside’: ’east’, ’westside’: ’west’, ’western’: ’west’, ’southside’: ’south’, ’southern’: ’south’
Time	Remove words as ’after’, ’before’ and etc., and sort to the ’hh:mm’ time format. ’X pm’ format is remained as the original.
Stars	[0-9]-stars, converted to [0-9]
Parking and Internet	’Free’ value for parking and internet slot is converted to ’yes’
Plural	’hotels’: ’hotel’, ’guesthouses’: ’guesthouse’, ’churches’: ’church’, ’museums’: ’museum’, ’entertainments’: ’entertainment’, ’colleges’: ’college’, ’nightclubs’: ’nightclub’, ’swimming pools’: ’swimming pool’, ’architectures’: ’architecture’, ’cinemas’: ’cinema’, ’boats’: ’boat’, ’boating’: ’boat’, ’theatres’: ’theatre’, ’concert halls’: ’concert hall’, ’parks’: ’park’, ’local sites’: ’local site’, ’hotspots’: ’hotspot’

Table 8: Value normalization rules when updating values from dialogue acts to dialogue states.

B SUMBT Slot Accuracy

slot type	MultiWOZ 2.1	MultiWOZ 2.2	MultiWOZ 2.3
attraction-area	0.959435626	0.95970696	0.962827296
attraction-name	0.93637227	0.939221273	0.952787953
attraction-type	0.967575634	0.971238638	0.965269299
hotel-area	0.943291277	0.944376611	0.94654728
hotel-book day	0.988739655	0.990638991	0.990367657
hotel-book people	0.986568987	0.987247321	0.989282323
hotel-book stay	0.992266992	0.994980328	0.99701533
hotel-internet	0.970153303	0.970153303	0.974494641
hotel-name	0.946682947	0.937593271	0.947089947
hotel-parking	0.970424637	0.971916972	0.978971646
hotel-pricerange	0.959978293	0.962284629	0.959028626
hotel-stars	0.978835979	0.979514313	0.979921313
hotel-type	0.946682947	0.942205942	0.959164292
restaurant-area	0.962962963	0.954687288	0.955229955
restaurant-book day	0.989010989	0.989146656	0.988332655
restaurant-book people	0.989146656	0.98982499	0.991724325
restaurant-book time	0.994301994	0.992402659	0.993080993
restaurant-food	0.976936644	0.976122643	0.974901642
restaurant-name	0.927146927	0.931759598	0.951024284
restaurant-pricerange	0.953601954	0.956450956	0.957536291
taxi-arriveBy	0.983584317	0.980328314	0.981820648
taxi-departure	0.961334961	0.96350563	0.961470628
taxi-destination	0.956993624	0.954958622	0.955636956
taxi-leaveAt	0.989146656	0.989553656	0.990367657
train-arriveBy	0.964048297	0.964048297	0.965404965
train-book people	0.972595306	0.970424637	0.97286664
train-day	0.986297653	0.986026319	0.990367657
train-departure	0.984262651	0.983991317	0.975579976
train-destination	0.985483652	0.98304165	0.97964998
train-leaveAt	0.93637227	0.941391941	0.93976394

Figure 6: Slot accuracies among MultiWOZ 2.1, MultiWOZ 2.2 and MultiWOZ 2.3 in terms of different slot types. The green color indicates the highest accuracy across all three datasets for a slot.

slot type	MultiWOZ 2.1	MultiWOZ 2.2	MultiWOZ 2.3-cof
attraction-area	0.959435626	0.95970696	0.967982635
attraction-name	0.93637227	0.939221273	0.945868946
attraction-type	0.967575634	0.971238638	0.969067969
hotel-area	0.943291277	0.944376611	0.950210284
hotel-book day	0.988739655	0.990638991	0.99321666
hotel-book people	0.986568987	0.987247321	0.991724325
hotel-book stay	0.992266992	0.994980328	0.99701533
hotel-internet	0.970153303	0.970153303	0.975579976
hotel-name	0.946682947	0.937593271	0.947089947
hotel-parking	0.970424637	0.971916972	0.98344865
hotel-pricerange	0.959978293	0.962284629	0.964048297
hotel-stars	0.978835979	0.979514313	0.980870981
hotel-type	0.946682947	0.942205942	0.956450956
restaurant-area	0.962962963	0.954687288	0.960520961
restaurant-book day	0.989010989	0.989146656	0.99660833
restaurant-book people	0.989146656	0.98982499	0.992131325
restaurant-book time	0.994301994	0.992402659	0.994573328
restaurant-food	0.976936644	0.976122643	0.976393976
restaurant-name	0.927146927	0.931759598	0.949124949
restaurant-pricerange	0.953601954	0.956450956	0.962555963
taxi-arriveBy	0.983584317	0.980328314	0.984533985
taxi-departure	0.961334961	0.96350563	0.974901642
taxi-destination	0.956993624	0.954958622	0.975851309
taxi-leaveAt	0.989146656	0.989553656	0.99023199
train-arriveBy	0.964048297	0.964048297	0.967575634
train-book people	0.972595306	0.970424637	0.97666531
train-day	0.986297653	0.986026319	0.993759327
train-departure	0.984262651	0.983991317	0.975037308
train-destination	0.985483652	0.98304165	0.978564645
train-leaveAt	0.93637227	0.941391941	0.939628273

Figure 7: Slot accuracies among MultiWOZ 2.1, MultiWOZ 2.2 and MultiWOZ 2.3-cof in terms of different slot types. The green color indicates the highest accuracy across all three datasets for a slot.