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

Ting Han\textsuperscript{1}∗ Ximing Liu\textsuperscript{2}∗ Ryuichi Takanobu\textsuperscript{3} Yixin Lian\textsuperscript{2} Chongxuan Huang\textsuperscript{2} Dazhen Wan\textsuperscript{3} Wei Peng\textsuperscript{2}† Minlie Huang\textsuperscript{3}†

\textsuperscript{1}University of Illinois at Chicago
\textsuperscript{2}Artificial Intelligence Application Research Center (AARC), Huawei Technologies
\textsuperscript{3}Tsinghua University

\{than24@uic.edu, aihuang@tsinghua.edu.cn\}
\{gxly19, wandz19\}@mails.tsinghua.edu.cn
\{liuximing1, huang.chongxuan, lianyixin1, peng.weil\}@huawei.com

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 dialogue 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 (Zhu et al., 2020a; Henderson et al., 2014; Williams et al., 2013; Wen et al., 2017; Rastogi et al., 2019; Budzianowski et al., 2018).

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 (Takanobu et al., 2019; Zhao et al., 2019), dialogue generation (Chen et al., 2019) and dialogue state tracking (Zhou and Small, 2019; Zhang et al., 2019; Heck et al., 2020; 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 or around 60%.\textsuperscript{1} MultiWOZ 2.1 (Eric et al., 2020) 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 (Zang et al., 2020) to improve annotation quality. This schema-based dataset contains annotations allowing for directly retrieving slot values from a given dialogue context (Zhang et al., 2019; Gao et al., 2019; Heck et al., 2020). Despite achieving a noticeable annotation quality uplift compared to that for the original MultiWOZ, there is still room 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\textsuperscript{2}. Our contributions are as follow:

- We differentiate incorrect annotations in dialogue acts from those in dialogue states, and unify annotations of dialogue acts and dialogue states to ensure their consistency when

\textsuperscript{1}Both authors contributed equally to the work. The work was conducted when Ting Han was interning at Huawei AARC.

\textsuperscript{2}Corresponding author.

\textsuperscript{1}https://github.com/budzianowski/multiwoz
\textsuperscript{2}https://github.com/lexmen318/MultiWOZ-coref

Marked date: 6-1-2021
Table 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.

| Error Type | Dialogue ID | Utterance | 2.1 Dialogue.act | 2.3 Dialogue.act |
|------------|-------------|-----------|------------------|------------------|
| Under-annotated | SSNG0348.json | For 3 people starting on Wednesday and staying 2 nights . | Hotel-Inform.Stay: 2 | Hotel-Inform.Stay: 2 |
| | PMUL1170.json | Yes , one ticket please , can I also get the reference number ? | Train-Inform.People: 1 | Train-Inform.People: one |
| | SNG01856.json | no, i just need to make sure it's cheap , oh, and i need parking | 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.Dest: arbury lodge guesthouse | Taxi-Inform.Leave: 4:45 Taxi-Inform.Dest: the college |
| | 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 , the 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 |
| 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 | Yes please book for that room for 2 nights . | Hotel-Inform.Price: cheap Hotel-Inform.Stay: 2 | Hotel-Inform.Stay: 2 |

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, Eric et al. (2020) re-annotated dialogue acts for both systems and users in MultiWOZ 2.1. We broadly categorize the incorrect annotations into three types (Table 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 (Zhu et al., 2020b) and merge them with original “dialog_act” annotations; b) we use assorted regular
**Dialogue ID** | **Utterance** | **MultiWOZ 2.1** | **MultiWOZ 2.3**
--- | --- | --- | ---
MUL2602.json | **User:** Can you recommend me a nightclub where I can get jiggly with it?  
**Sys:** Well, I think the jiggliest nightclub in town is the Soul Tree Nightclub, right in centre city! Plus the entrance fee is only 4 pounds | a-type=nightclub  
a-name=not mentioned  
a-area=not mentioned | a-type=nightclub  
a-name=not mentioned  
a-area=not mentioned

MUL1455.json | **User:** That is perfect can I have the postcode please?  
**Sys:** Sure! The postcode is cb23qf | 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

Table 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.

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 3 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 3: A comparison of annotation correctness ratios of “dialog_act” for MultiWOZ 2.1/2.2 and coref. 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 Table 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” annotations), 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 (Table 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. To this end, we follow the way how MultiWOZ 2.1 normalizes slot values based on utterances.

- **Consistent Tracking Strategy:** The inconsistent tracking strategy (Figure 1) was initially discussed (but not solved) in MultiWOZ 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
the user to the system, the informed values are propagated to the next turn of dialogue states; b) if an informing/recommending action is from the system to the user, 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 4: Percentage of slots’ values changed in MultiWOZ 2.3 and MultiWOZ 2.1, respectively, for “metadata” annotations. “Value Filled” stands for a value-filled from null, “none” or “not mentioned”. “Value Changed” means a slot value is changed to “not mentioned” or null. “Value dontcare” stands for slot values filled with “dontcare”.

3 Enhance Dataset with Co-referencing

MultiWOZ contains a considerable amount of co-reference and ellipsis. As shown in Table 5, 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 enhanced with co-referencing features to achieve SOTA results (Ferreira Cruz et al., 2020). By including co-reference in CamRest676 (Wen et al., 2017), GECOR (Quan et al., 2019) showed significant performance improvement compared to the baseline models. Through restoring incomplete utterances by annotating the dataset with co-reference labels, Pan et al. (2019) 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 6. In total, 20.16% dialogues are annotated with co-reference in the dataset, indicating the importance of co-referencing annotation.

3.1 Annotation for Co-reference in Dialogue

The “coreference” annotations are applied to all “dialog_act” slots having co-referencing relationships
Table 5: Examples of co-reference annotations. Co-reference values are added to the original utterances and marked as light orange italic inside the brackets.

| 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 6: Statistics of co-reference annotations. H/R/A/T represent “Hotel”, “Restaurant”, “Attraction” and “Train”, respectively.

| 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% |

3.2 Annotation for Co-reference in User Goal

During the data collection process, the user converses with the system, following a given goal description (Budzianowski et al., 2018). Co-reference in the user utterances is derived from co-reference in user goals. However, the goal annotation, represented as several constraints and requests, is not consistent with the goal description and does not implement co-reference features. Table 7 shows two examples of user goals with co-reference. The original goal annotation misses a request, three constraints and all co-reference relations. The right arrow (hotel.stay=3→1) indicates a possible goal change during a dialogue. The co-referencing relations are represented as referenced domains and slots. Note that the referenced slot of “taxi.departure/taxi.destination” is uncer-

Figure 2: 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 back to the history utterances where the co-referred slots are first encountered;
- We use the corresponding dialogue acts with paired span information to retrieve co-referred values.

In total, we added 3,340 co-referencing annotations for “dialog_act”.

Table 7 shows examples of user goals with co-reference. The right arrow (hotel.stay=3→1) indicates a possible goal change during a dialogue. The co-referencing relations are represented as referenced domains and slots. Note that the referenced slot of “taxi.departure/taxi.destination” is uncer-

| 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 6 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”.

3.2 Annotation for Co-reference in User Goal

During the data collection process, the user converses with the system, following a given goal description (Budzianowski et al., 2018). Co-reference in the user utterances is derived from co-reference in user goals. However, the goal annotation, represented as several constraints and requests, is not consistent with the goal description and does not implement co-reference features. Table 7 shows two examples of user goals with co-reference. The original goal annotation misses a request, three constraints and all co-reference relations. The right arrow (hotel.stay=3→1) indicates a possible goal change during a dialogue. The co-referencing relations are represented as referenced domains and slots. Note that the referenced slot of “taxi.departure/taxi.destination” is uncer-

In total, we added 3,340 co-referencing annotations for “dialog_act”.

Table 6: Statistics of co-reference annotations. H/R/A/T represent “Hotel”, “Restaurant”, “Attraction” and “Train”, respectively.
You are looking for a place to stay. The hotel should include free parking and should be in the same price range as the restaurant. The hotel should include free wifi. Once you find the hotel, you want to book it for the same group of people and 3 nights starting from the same day. If the booking fails how about 1 nights. Make sure you get the reference number.

Constraint
hotel.parking=yes
hotel.pricerange=expensive
hotel.internet=yes
hotel.people=3
hotel.day=wednesday
hotel.stay=3

Request
hotel.Ref=?

You also want to book a taxi to commute between the two places. You want to leave the attraction by 02:45. Make sure you get contact number and car type.

Constraint
taxi.leaveAt=02:45
Request
taxi.phone=?
taxi.car type=?

Table 7: Examples of co-reference annotations in the user goal. The red color highlights the difference between the original and new annotations.

Table: 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 (Zhu et al., 2020b) 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 (Chen et al., 2019), 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 8 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 (Zhang et al., 2019; Heck et al., 2020; 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 up-

---

Footnote:

3Full benchmarks with various models are available in Appendix B.
Table 8: 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    | 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 9: 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.

| 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 8: Performance of SUMBT and TRADE over different versions of dataset. MultiWOZ-coref 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 8 and 10, 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 adopts a rule-based method to correct the identified errors in MultiWOZ 2.1 further. With a customized pre-process script for SUMBT, the joint goal accuracy can reach 54.54% for MultiWOZ 2.3 and 56.09% for MultiWOZ-coref, respectively. 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 3 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 Table 13 at Appendix C. As

4 Scores shown in Table 10 are achieved by using pre-process scripts provided by SUMBT and TRADE.
shown in Table 13, our dataset achieves the best performance for 17 out of all 30 slots. The performance is further enhanced with the co-reference version (24 out of all 30).

Table 9 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 11 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, 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 dialogue 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

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.

Qian Chen, Zhu Zhuo, and Wen Wang. 2019. BERT for Joint Intent Classification and Slot Filling. arXiv e-prints, page arXiv:1910.10909.

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.

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyang Gao, Adarsh Kumar, Anuj Goyal, Peter Ku, and Dilek Hakkani-Tur. 2020. MultiWOZ 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. In Proceedings of The 12th Language Resources and Evaluation Conference, pages 422–428, Marseille, France. European Language Resources Association.

André Ferreira Cruz, Gil Rocha, and Henrique Lopes Cardoso. 2020. Coreference resolution: Toward end-to-end and cross-lingual systems. Information, 11(2):74.

Shuyang Gao, Abhishek Sethi, Sanchit Agarwal, Taisyong Chung, and Dilek Hakkani-Tur. 2019. Dialog state tracking: A neural reading comprehension approach. In Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue, pages 264–273, Stockholm, Sweden. Association for Computational Linguistics.

Izzeddin Gür, Dilek Hakkani-Tür, Gokhan Tür, and Pararth Shah. 2018. User modeling for task oriented dialogues. In 2018 IEEE Spoken Language Technology Workshop (SLT), pages 900–906. IEEE.

Michael Heck, Carel van Niekerk, Nurul Lubis, Christian Geishauser, Hsien-Chin Lin, Marco Moresi, and Milica Gasic. 2020. TripPy: A triple copy strategy for value independent neural dialog state tracking.
In Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue, pages 35–44, 1st virtual meeting. 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.

Sungdong Kim, Sohee Yang, Gyuwan Kim, and Sang-woo Lee. 2020. Efficient dialogue state tracking by selectively overwriting memory. In ACL.

Hwaran Lee, Jinsik Lee, and Tae-Yoon Kim. 2019a. Zhufeng Pan, Kun Bai, Yan Wang, Lianqiang Zhou, Shikib Mehri, Mihail Eric, and Dilek Hakkani-Tur. 2019b. Sungjin Lee, Qi Zhu, Ryuichi Takanobu, Zheng Zhang, Yaqin Zhang, Xiang Li, Jinhao 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, pages 5478–5483, Florence, Italy. Association for Computational Linguistics.

Shikib Mehri, Mihail Eric, and Dilek Hakkani-Tur. 2020. Dialoglue: A natural language understanding benchmark for task-oriented dialogue.

Zhufeng Pan, Kun Bai, Yan Wang, Lianqiang Zhou, and Xiaojiang Liu. 2019. Improving open-domain dialogue systems via multi-turn incomplete utterance restoration. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 1824–1833, Hong Kong, China. Association for Computational Linguistics.

Jun Quan, Deyi Xiong, Bonnie Webber, and Changjian Hu. 2019. GECOR: An end-to-end generative ellipsis and co-reference resolution model for task-oriented dialogue. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 4547–4557, Hong Kong, China. Association for Computational Linguistics.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghab Gupta, and Pranav Khaitan. 2019. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. arXiv preprint arXiv:1909.05855.

Liliang Ren, Jianmo Ni, and Julian McAuley. 2019. Scalable and accurate dialogue state tracking via hierarchical sequence generation. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 1876–1885, Hong Kong, China. Association for Computational Linguistics.

Jost Schatzmann, Blaise Thomson, Karl Weilhammer, Hui Ye, and Steve Young. 2007. Agenda-based user simulation for bootstrapping a pomdp dialogue system. In Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics: Companion Volume, Short Papers, pages 149–152.

Hui Su, Xiaoyu Shen, Rongzhi Zhang, Fei Sun, Pengwei Hu, Cheng Niu, and Jie Zhou. 2019. Improving multi-turn dialogue modeling 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.

Ryuichi Takanobu, Hanlin Zhu, and Minlie Huang. 2019. Guided dialog policy learning: Reward estimation for multi-domain task-oriented dialog. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 100–110, Hong Kong, China. Association for Computational Linguistics.

Yexiang Wang, Yi Guo, and Siqi Zhu. 2020. Slot attention with value normalization for multi-domain dialogue state tracking. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 3019–3028, Online. Association for Computational Linguistics.

Tsung-Hsien Wen, David Vardyke, 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.

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, Elhsan 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.
Xiaoxue Zang, Abhinav Rastogi, and Jindong Chen. 2020. MultiWOZ 2.2: A dialogue dataset with additional annotation corrections and state tracking baselines. In Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI, pages 109–117, Online. Association for Computational Linguistics.

Jian-Guo Zhang, Kazuma Hashimoto, Chien-Sheng Wu, Yao Wan, Philip S Yu, Richard Socher, and Caiming Xiong. 2019. Find or classify? dual strategy for slot-value predictions on multi-domain dialog state tracking. arXiv preprint arXiv:1910.03544.

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.

Li Zhou and Kevin Small. 2019. Multi-domain dialogue state tracking as dynamic knowledge graph enhanced question answering. arXiv preprint arXiv:1911.06192.

Qi Zhu, Kaili Huang, Zheng Zhang, Xiaoyan Zhu, and Minlie Huang. 2020a. Crosswoz: A large-scale chinese cross-domain task-oriented dialogue dataset. arXiv preprint arXiv:2002.11893.

Qi Zhu, Zheng Zhang, Yan Fang, Xiang Li, Ryuichi Takanobu, Jinchao Li, Baolin Peng, Jianfeng Gao, Xiaoyan Zhu, and Minlie Huang. 2020b. ConvLab-2: An open-source toolkit for building, evaluating, and diagnosing dialogue systems. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pages 142–149, Online. Association for Computational Linguistics.
A Value Normalization

| 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 11: Value normalization rules when updating values from dialogue acts to dialogue states.

B Dialogue State Tracking benchmarks

Upon code availability, we experiment MultiWOZ 2.3 on various dialogue state tracking models and Table 12 shows the corresponding joint goal accuracies.

| Models          | MultiWOZ 2.1 | MultiWOZ 2.3 |
|-----------------|--------------|--------------|
| TRADE (Wu et al., 2019) | 45.6%        | 49.2%        |
| SUMBT (Lee et al., 2019a) | 49.2%        | 52.9%        |
| COMER (Ren et al., 2019) | 48.8%        | 50.2%        |
| DSTQA (Zhou and Small, 2019) | 51.2%        | 51.8%        |
| SOM-DST (Kim et al., 2020) | 53.1%        | 55.5%        |
| TripPy (Heck et al., 2020) | 55.3%        | 63.0%        |
| ConvBERT-DG-Multi (Mehri et al., 2020) | 58.7%        | 67.9%        |
| SAVN (Wang et al., 2020) | 54.5%        | 58.0%        |

Table 12: Joint goal accuracies for different dialogue state tracking models on the MultiWOZ 2.1 and MultiWOZ-coref. We notice our work is cocurrent with MultiWOZ 2.2. However, we mainly base our refinement on MultiWOZ 2.1 and many models do not report joint goal accuracies on MultiWOZ 2.2. Therefore, MultiWOZ 2.2 is excluded from comparison.

C SUMBT Slot Accuracy
| Slot type         | MultiWOZ 2.1 | MultiWOZ 2.2 | MultiWOZ 2.3 | MultiWOZ-coref |
|------------------|--------------|--------------|--------------|----------------|
| attraction-area  | 95.94        | 95.97        | **96.28**    | **96.80**      |
| attraction-name  | 93.64        | 93.92        | **95.28**    | **94.59**      |
| attraction-type  | 96.76        | **97.12**    | 96.53        | 96.91          |
| hotel-area       | 94.33        | 94.44        | **94.65**    | **95.02**      |
| hotel-book day   | 98.87        | 99.06        | 99.04        | **99.32**      |
| hotel-book people| 98.66        | 98.72        | **98.93**    | **99.17**      |
| hotel-book stay  | 99.23        | 99.50        | **99.70**    | **99.70**      |
| hotel-internet   | 97.02        | 97.02        | **97.45**    | 97.56          |
| hotel-name       | 94.67        | 93.76        | **94.71**    | **94.71**      |
| hotel-parking    | 97.04        | 97.19        | **97.90**    | **98.34**      |
| hotel-pricerange | 96.00        | 96.23        | 95.90        | **96.40**      |
| hotel-stars      | 97.88        | 97.95        | **97.99**    | **98.09**      |
| hotel-type       | 94.67        | 94.22        | **95.92**    | **95.65**      |
| restaurant-area  | **96.30**    | 95.47        | 95.52        | 96.05          |
| restaurant-book day | 98.90    | 98.91        | 98.83        | **99.66**      |
| restaurant-book people | 98.91 | 98.98        | **99.17**    | **99.21**      |
| restaurant-book time | 99.43  | 99.24        | 99.31        | **99.46**      |
| restaurant-food  | **97.69**    | 97.61        | 97.49        | 97.64          |
| restaurant-name  | 92.71        | 93.18        | **95.10**    | **94.91**      |
| restaurant-pricerange | 95.36 | 95.65        | **95.75**    | **96.26**      |
| taxi-arriveBy    | 98.36        | 98.03        | 98.18        | **98.45**      |
| taxi-departure   | 96.13        | 96.35        | 96.15        | **97.49**      |
| taxi-destination | 95.70        | 95.50        | 95.56        | **97.59**      |
| taxi-leaveAt     | 98.91        | 98.96        | **99.04**    | **99.02**      |
| train-arriveBy   | 96.40        | 96.40        | **96.54**    | **96.76**      |
| train-book people | 97.26    | 97.04        | **97.29**    | **97.67**      |
| train-day        | 98.63        | 98.60        | **99.04**    | **99.38**      |
| train-departure  | **98.43**    | 98.40        | 97.56        | 97.50          |
| train-destination | **98.55** | 98.30        | 97.96        | 97.86          |
| train-leaveAt    | 93.64        | **94.14**    | 93.98        | 93.96          |

Table 13: Slot accuracies among MultiWOZ 2.1, MultiWOZ 2.2, MultiWOZ 2.3 and MultiWOZ-coref in terms of different slot types. The bold number indicates the highest accuracy across all three datasets for each slot. The red bold number indicates higher accuracy between MultiWOZ 2.3 and MultiWOZ-coref for each slot.