A Slot Is Not Built in One Utterance: Spoken Language Dialogs with Sub-Slots

Sai Zhang 1, Yuwei Hu 1, Yuchuan Wu 2, Jianman Wu 2, Yongbin Li 1, Jian Sun 1, Caixia Yuan 1 and Xiaojie Wang 1

1Beijing University of Posts and Telecommunications, Beijing, China
2Independent Researcher

{zs, hyw724, yuanxcx, xjwang}@bupt.edu.cn
jw.0123@outlook.com, jiansun_china@hotmail.com
liyb821@gmail.com

Abstract

A slot value might be provided segment by segment over multiple-turn interactions in a dialog, especially for some important information such as phone numbers and names. It is a common phenomenon in daily life, but little attention has been paid to it in previous work. To fill the gap, this paper defines a new task named Sub-Slot Based Task-Oriented Dialog (SSTOD) and builds a Chinese dialog dataset SSD for boosting research on SSTOD. The dataset includes a total of 40K dialogs and 500K utterances from four different domains: Chinese names, phone numbers, ID numbers and license plate numbers. The data is well annotated with sub-slot values, slot values, dialog states and actions. We find some new linguistic phenomena and interactive manners in SSTOD which raise critical challenges of building dialog agents for the task. We test three state-of-the-art dialog models on SSTOD and find they cannot handle the task well on any of the four domains. We also investigate an improved model by involving slot knowledge in a plug-in manner. More work should be done to meet the new challenges raised from SSTOD which widely exists in real-life applications. The dataset and code are publicly available via https://github.com/shunjiu/SSTOD.

1 Introduction

Task-oriented dialogs help users accomplish specific tasks such as booking restaurants or accessing technical support services by acquiring task-related slots through multi-turn dialogs. Many advances have been achieved under an assumption that each slot value is informed or updated as a whole in a single turn by default (Li et al., 2017; Zhang et al., 2020b; Hosseini-Asl et al., 2020; Dai et al., 2021). But in real-world dialogs, some slot values are often provided in a much more complicated manner.

We take phone numbers as an example. Users tend to inform an agent a sequence of 0-9 digits segment by segment across several turns as exemplified in Figure 1. Accordingly, the agent needs to confirm, update or record the recognized sub-slot values. We regard these scenarios as SSTOD task.

The SSTOD is very common when people communicate telephone numbers, names and so on. Specifically, as shown in Figure 1, the SSTOD task raises several critical new challenges which have not been tackled in building dialog agents: (1) Multi-segment informing: The segments could be informed in many different complex ways. As exemplified in Figure 1, the user informed two sub-slots “136” and “361555” sequentially. The agent needs to confirm, update or record the recognized sub-slot values. We regard these scenarios as SSTOD task.

(2) Sub-slot locating: Differing from updating a whole slot value in traditional slot filling, in SSTOD, the agent needs to precisely locate the part of values that needs to be updated. The situation is exacerbated when there are more than one similar sub-slots. (3) Knowledge-rich relevancy: To avoid the ambiguities of speech, users usually introduce a piece of knowledge along with informing the slot values (Tsai et al., 2005; Wang, 2007). For example, the knowledge, “明天的明” is used to disambiguate character “明” (It is the similar case when
English speakers say “A as in Alpha” in phone calls). The agent should look into the knowledge in order to correct correct value.

To the best of our knowledge, the existing dialog benchmarks, such as ATIS (Hemphill et al., 1990), MultiWOZ (Budzianowski et al., 2018), Cross- WOZ (Zhu et al., 2020), and SGD (Rastogi et al., 2020) do not contain the dialogs illustrated in Figure 1, which makes the dialog agents optimized on them fail dramatically at conversing in sub-slot dialogs. To address the above challenges, we develop the Sub-slot Dialog (SSD) dataset which contains most popular sub-slot dialog scenarios including phone numbers (a sequence of digits 0-9), ID numbers (much longer digit sequence), person names (a sequence of Chinese characters), and license plate numbers (a mix of Chinese characters, digits and English letters). The dataset is originated from the real-world human-to-human conversations, then richly labeled and reprocessed by crowdsourcing. Although the dataset is in Chinese, the development methodology depicted in this work is also applicable to other languages.

Under the setting of SSTOD, we present an improved model, UBAR+, on the basis of UBAR (Yang et al., 2021) and the large pretrained model GPT2 (Radford et al., 2018). UBAR+ equips UBAR with a knowledge prediction module to correct Automatic Speech Recognition (ASR) errors and discriminate the ambiguities, and achieves better performance on SSD. We also provide a rule-based user simulator to evaluate the system.

Our main contributions are:

- We propose a novel sub-slot based dialog task which exists widely in real-world conversations but has been neglected in previous work.

- We build a large-scale high-quality spoken Chinese dataset SSD for SSTOD, covering four common scenarios including phone numbers, ID numbers, Chinese names and license plate numbers collection, which will essentially benefit future research on SSTOD.

- We design a knowledge prediction module together with knowledge retrieval which helps UBAR achieve significant improvement on the name domain. Otherwise, a user simulator is provided to facilitate the evaluation of the system.

2 Task and Dataset

We first give a definition of SSTOD, then introduce how to build the SSD dataset, and give some analyses on the dataset.

2.1 Task Definition

We proposed sub-slot based dialog system as a one slot filling task. A user may provide a slot via multiple turns in oral conversations. In each turn, only a piece of the value, which is regarded as a sub-slot, is given. It is because the values like phone numbers are usually too long for a user to inform in one turn or the segments in values like surnames in names are often accompanied with extra explanations to disambiguate homonyms.

2.2 Dataset Creation

Since information such as phone numbers and names is private, real data cannot be used directly. We design a semi-automatic method to obtain a large-scale high-quality dialog dataset while avoiding privacy issues. We build a dataset in four domains including Phone Number, Name, ID Number and License Plate Number. We demonstrate the building process of the dataset by taking Name as an example.

| System action | request, continue, req more, implicit confirm, explicit confirm, ack, req correct, compare, ask restart, bye, how signal, good signal, robot, other |
|--------------|----------------------------------------------------------------------------------------------------------------------------------|
| User action  | offer, inform, update, affirm, deny, ack, ask state, restart, ask repeat, finish, wait, doubt identity, how signal, bad signal, good signal, other |

|                     | inform | update | affirm | deny | ack | ask state | restart |
|---------------------|--------|--------|--------|------|-----|-----------|---------|
| request             | 0.89   | 0.00   | 0.00   | 0.04 | 0.00|0          |0        |
| req more            | 0.93   | 0.00   | 0.00   | 0.00 | 0.00|0          |0        |
| implicit confirm    | 0.49   | 0.16   | 0.25   | 0.07 | 0.00|0          |0.01     |
| explicit confirm    | 0.00   | 0.30   | 0.66   | 0.00 | 0.00|0          |0.01     |
| ack                 | 0.94   | 0.00   | 0.00   | 0.04 | 0.02|0          |0        |
| compare             | 0.60   | 0.39   | 0.00   | 0.00 | 0.00|0          |0        |
| ask restart         | 0.00   | 0.00   | 0.15   | 0.33 | 0.50|0          |0        |

Figure 2: All actions in the phone domain (above) and part of transition probabilities (below). Each row in the table below is the probability of user action when a system action is given.

Human-to-Human (H2H) dialog. We sample 47,252 H2H dialogs from a business service by considering different time of service and different genders of customers, and obtain 4,489, 8,873 and 5,827 fragments of dialog for phone numbers, names and license plate numbers respectively. We analyze the H2H dialogs carefully, summarize some dialog actions and dialog policy, and estimate the transition probabilities between different
actions. Taking phone numbers as an example, we have 30 actions. Figure 2 gives part of transition probabilities between those actions.

Knowledge Base. Chinese characters in names cannot be disambiguated by context in spoken conversations. For example, when someone says, “我姓吴 (my surname is Wu)”, different Chinese characters which share the same pronunciation of “wu”, including “吴”, “武”, “伍”, etc., are all possible to be the surname to the listeners. People therefore always employ some external knowledge to distinguish different characters. For example, “我姓吴,口天吴 (my surname is ‘吴’, ‘口’ and ‘天’ compose ‘吴’)”, where “口天吴” is a piece of external knowledge. It gives components (normally some simple characters) of a character. People also use knowledge of character combination (i.e. words or phrases) to identify a Chinese character. For example, “我姓吴,东吴的吴 (my surname is ‘吴’, ‘吴’ as in ‘DongWu’) ”, where ‘DongWu’ is a word which only “吴” fits the word well. “DongWu” is another piece of knowledge for Chinese character “吴”. Almost all frequent Chinese Characters have several pieces of knowledge as above. Appendix A gives some pieces of knowledge on Chinese characters. Knowledge is widely used in name telling. We thus build 20,547 pieces of knowledge for 2,003 common used Chinese characters. On average, each Chinese character is with more than 10 pieces of knowledge. We give more examples in Appendix A.

Data generation. Based on the analysis of H2H dialogs, two probabilistic FSA-based simulators are built for System and User respectively, both with a template-based Nature Language Generation (NLG) module for generating natural language sentences from actions sampled from probabilistic FSA. We give some examples of NLG modules in Appendix B. Part of FSAs is given in Appendix C. An error simulator is also built for modeling errors brought by ASR. Two FSAs as well as a NLG module and an error model work together to generate various dialogs. At the beginning, the FSA for users initializes a target slot value which is composed of several sub-slot segments. The two probabilistic FSAs then interact based on the sampled actions. At each step, when FSA chooses current dialog action and sub-slot values, a NLG template is randomly chosen to generate a sentence. The error model might also be triggered randomly to twist the values with a defined probability. When the system thinks it collects a complete slot value, it ends the dialog. If the slot value collected is consistent with the slot value initialized by the user, the dialog succeeds; otherwise, the dialog fails. Appendix D illustrates several example dialogs generated by FSAs.

Data crowdsourcing. To make our dialog data more natural and diverse, we hired crowd workers to paraphrase user utterances in the generated dialogs. New utterances bring more templates, knowledge pieces and real ASR errors. Table 1 gives the numbers of crowdsourced data.

| Domains→ | PHONE | ID     | NAME | PLATE |
|----------|-------|--------|------|-------|
| Types    | 8,578 | 7,350  | 3,031| 5,179 |
| Templates| 3,849 | -      | 29,874| 10,000|
| Sentences| -     | -      | 34,302| -     |
| Knowledge| -     | -      | -    | -     |

Table 1: Numbers of crowdsourced data.

Figure 3: The distribution of numbers of sentences in a dialog (left) and the distribution of numbers of characters in a sentence (right).
Table 2: Analysis of the SSD dataset.

|                         | SSD-PHONE | SSD-ID | SSD-NAME | SSD-PLATE |
|-------------------------|-----------|--------|----------|-----------|
| No. of dialogs          | 11,000    | 8,000  | 15,000   | 6,000     |
| No. of actions          | 30        | 30     | 29       | 27        |
| Avg. turns per dialog   | 13.01     | 16.86  | 9.86     | 13.90     |
| Avg. tokens per sentence| 11.61     | 13.13  | 7.70     | 13.84     |
| Avg. sub-slots per dialog| 2.90    | 4.15   | 2.84     | 2.03      |
| No. of different paths  | 3,135     | 5,412  | 2,475    | 3,965     |
| Vocabulary size         | 677       | 629    | 3,519    | 915       |

2.3 Data Statistics

We finally obtained a large and high-quality data for SSTOD in four domains. Some statistics are shown in Table 2.

As we can see in Table 2, the SSD dataset has 40K dialogs and the number of dialogs exceeds that of most available task-oriented datasets (the largest dialog dataset SGD (Rastogi et al., 2020) commonly used today contains 16,142 dialogs). The number of actions is at least 27 in each domain, which is more than that in any single domain of the currently commonly used dataset MultiWOZ (Budzianowski et al., 2018).

The average turn per dialog is no less than 10, as well as the average character per sentence. The distribution of dialog length is shown in Figure 3 (left) and the distribution of dialog sentence length per domain is shown in Figure 3 (right).

A path is the action sequence in a dialog. Two dialogs with distinct paths means they have different ways to complete a task. The larger the number of different paths, the more diversity of action sequences. The SSD dataset shows adequate diversity of dialogs.

The average number of sub-slots per dialog is the average number of pieces that a full slot value is segmented. It can be seen that names are averagely segmented into 2.84 pieces. Considering a Chinese name normally includes 2-3 Chinese characters, people say their names character by character.

Finally, it should be noticed that data contains a wealth of annotation information. For each user utterance, we annotate an action and the sub-slot values provided by the user. For each system utterance, we annotate an action and the state which is the sub-slot value collected by the system. The annotation information allows our data to be used for the following tasks: natural language understanding (NLU), dialog state tracker (DST), dialog policy, NLG, etc. We will also release our FSA-based User simulator, which can be used to evaluate the system.

2.4 New Challenges

The dataset includes many new phenomena that are seldom seen in other datasets which bring some new challenges to build agents for SSTOD. Most of the new phenomena are brought by the sub-slot telling way. Table 3 gives some of these new phenomena as well as a sample utterance for each phenomenon.

Most of the phenomena listed in Table 3 are seldom seen in the previous dialog datasets. They raise some new challenges on at least three sides: The first one is to locate and record each segment and even each element in each segment, since all of them might be updated separately or as a whole. The second one is to identify the various external knowledge, especially when ASR errors are involved. The third one is that the context of the sub-slot might be helpless when there are ambiguities. The knowledge might be the major source of disambiguation, including those explicitly noticed in utterances, as well as implicitly used in dialogs.

3 Method

3.1 Benchmark Models

Since the new task raises critical challenges, we firstly verify whether the current state-of-the-art (SOTA) models on normal task-oriented dialog task can meet the challenges, then we take a small step on improving one SOTA model by introducing a specific plug-in component to make it handle some of the challenges.

Recently, many strong models have been proposed to tackle the MultiWOZ benchmark (Hosseini-Asl et al., 2020; Yang et al., 2021; He et al., 2022). In this paper, we chose three SOTA dialog models for our SSTOD evaluation as follows:
Table 3: Part of the diversity cases and their examples.

| Description                                                                 | Example                                                                 |
|----------------------------------------------------------------------------|-------------------------------------------------------------------------|
| Inform (quantifier)                                                        | 1. 4个3 (1, four 3’s.)                                                 |
| Inform (correct)                                                           | 嗯1820，嗯，不是是1860 (Uh-huh1820, hmm, no it’s 1860.)                |
| Inform (repeat)                                                            | 7127 7127                                                              |
| Inform (stretched)                                                         | 1. 1044                                                                |
| Inform (overlap)                                                           | User: 嗯，您那麻麻，您记一下的手机号码。181 (Well, would you mind writing down the phone number? 181.) |
|                                                                             | System: 嗯。181 (Uh-huh, 181.)                                           |
|                                                                             | User: 1814104                                                           |
| Update (refer)                                                             | 最后一位是5664 (The last 4 digits are 5664.)                           |
| Update (delete)                                                            | 去掉7 (Delete 7.)                                                      |
| Update (add)                                                               | 9后面少了个4 (Behind 9, 4 is missing.)                                 |
| Update (part)                                                              | System: 133 4777 3029. 好，我知道了，谢谢啊(133 4777 3029, ok, I see. Thanks!) |
|                                                                             | User: 529才对 (It is 529.)                                             |
| Sub-slot update                                                            | 2不对啊，是R，RST里面的R才对 (2 is not right, it’s R as in RST.)      |
|                                                                             | (note: 2 and R have the same pronunciation in Chinese.)                 |
| Comparison of homophonic characters                                        | 并且E还是数字1? ( Is it the letter E or number 1? ) (note: “E” and “1” have the same pronunciation in Chinese.) |
| Using external knowledge (character combination)                          | 艳是艳丽的艳 (“艳” is from “艳丽”, a two-character word means showy.)  |
| Using external knowledge (structure)                                       | 艳是一个单字，一个单字 (“艳” is composed of “单” and “字”.)            |
| ASR errors of a character or(and) its knowledge                           | ASR outputs: 驱是严厉的严，一个单字，一个单字 |
|                                                                             | Original utterance: 艳是艳丽的艳，一个单字，一个单字 |
|                                                                             | (“驱”和“严” are badly recognized characters of “艳”．“风” is a |
|                                                                             | badly recognized character of “风”，“艳” (showy) is the |
|                                                                             | correction of “严厉” (servere).)                                       |
| Two identical characters in one name                                       | 我叫李壮壮，状是状元的状，两个状都是 (My name is “李壮壮” (Li Zhuangzhuang), the last two words are both “状” as in “状元” (top students.) |
| Two characters from one knowledge                                          | 我叫严丰，亚精干勤的亚勤 (My name is “亚勤” (Ye Qin) as in Chinese idioms “亚勤” (Excellence in work lies in diligence).) |

TRADE (Wu et al., 2019) utilizes the generative approach and copy-generator mechanism for slot filling tasks. We construct a complete dialog system using TRADE and a rule-based policy module as a baseline.

SimpleTOD (Hosseini-Asl et al., 2020) uses a single, causal language model to aggregate dialog state tracking, policy deciding, and response generating a cascaded generator. Leveraging the large pre-trained model such as GPT2, SimpleTOD achieved competitive results on MultiWOZ.

UBAR (Yang et al., 2021) presents variants on Ham et al. (2020); Peng et al. (2020); Zhang et al. (2020a) to parameterize the dialog system as an auto-regressive model. It models the task-oriented dialog system on a dialog session level, instead of using all user and system utterances as inputs. Conditioned on all previous belief state, system acts and response, UBAR is easier to make inference and planning in current turn and achieves the state-of-the-art performance on MultiWOZ.

3.2 Plug-in Module

As described above, one of the challenges in SSD is that the disambiguation of the slot values intensely relies on both the context and the extra knowledge. For example, users might inform a person name by making use of character knowledge to distinguish the target characters from alternatives.

We therefore design a simple plug-in unit to execute Knowledge Prediction (KP) and Knowledge Retrieve (KR) on demand. Taking UBAR as a testbed, we proposed a UBAR with the plug-in unit (hereafter UBAR+) whose framework is illustrated in Figure 4.

Given a user input utterance $U_t$, $UBAR^+$ first generates knowledge snippets $K_t = [k_1^t, \ldots, k_m^t] \subset U_t$, where $m$ is the number of extracted snippets. Each snippet corresponds to a target sub-slot value. For instance, if utterance $U_t=$“我叫张艳，张是弓长张，艳是严厉的艳”， the extracted knowledge snippets $K_t = [k_1^t, k_2^t]$ = [“弓长张”，“严厉的艳”].

Both extracted knowledge snippets and the knowledge items in extra knowledge base are embedded via TF-IDF (Jones, 1972) vectors both in
char-level and pinyin-level (which is the phonetic transcription of a Chinese character).

Finally, the cosine similarities between the snippet $k^t_i \in K_t$ and each candidate knowledge item $kd_j$ from the knowledge base, are calculated as follows:

$$e_c(k^t_i) = \text{TF-IDF}_{\text{char}}(k^t_i),$$  
$$e_p(k^t_i) = \text{TF-IDF}_{\text{pinyin}}(k^t_i),$$

$$\text{score}(k^t_i, kd_j) = \alpha \cos (e_c(k^t_i), e_c(kd_j)) + (1 - \alpha) \cos (e_p(k^t_i), e_p(kd_j)),$$

where $e_c(k^t_i)$ and $e_p(k^t_i)$ have the length of vocabulary size of characters and pinyin, respectively.

For knowledge item $kd_h$ with the maximum similarity score, its corresponding character $w_k$ is used as the disambiguated character of $k^t_i$, yielding the predicted target sub-slot sequence $C_t = [w_1, \ldots, w_m]$.

Here we finish the disambiguation of one sub-slot value. By repeating the above procedures, all sub-slots are assigned their predicted target char, thereby the belief state (BS in Figure 4) is updated accordingly. To rationally navigate the following dialog, the agent then learns to plan its following acts of whether confirming a sub-slot or continuously requesting a sub-slot. We apply cross-entropy and language modeling objective (Bengio et al., 2003) to optimize the plug-in unit:

$$L_{\text{plug-in}} = \sum \log P(w_t|w_{<t}).$$

$L_{\text{plug-in}}$ is added to the loss applied in UBAR, making the final loss of the UBAR$^+$.

### 4 Experiments

Using the SSD dataset as a dialog state tracking benchmark, we conduct a comprehensive analysis of the challenges through an empirical approach and validate the effectiveness of the proposed UBAR$^+$ method.

#### 4.1 Experimental Setup

**Dataset.** We split the SSD dataset into a training set, a validation set and a test set in the ratio of 7:1:2 on each of the four domains and conduct experiments on them.

**Evaluation Metrics.** We evaluate model performances on SSD with several popularly used metrics. **Joint acc** is the accuracy of all sub-slot values at each turn. The output is considered as an accurate one if and only if all the sub-slot values are exactly consistent with the ground truth values. **Slot acc** measures whether each sub-slot is correctly collected at each turn. **Dialog succ** measures whether the collected slot value is consistent with the user’s goal at the end of the dialog. To have a comprehensive comparison, we also test our model by online interacting with FSA-based user simulators with two evaluation metrics: **Dialog succ** and **Avg turn**. **Dialog succ** is the main metric, which means the ratio of successful dialogs. A dialog is successful if and only if the slot is correctly collected by system within limited turns. **Avg turn** is used to measure the average turn number of successful dialogs.

**Implementation Details.** We initialize our proposed UBAR$^+$ model with ClueCorpus-small (Xu et al., 2020) and fine-tune it on SSD. The max length of an input sequence is set to 1024 and the excess parts are truncated. The $\alpha$ in the plug-in...
We implement three different evaluations on model performances: the first one is offline test where models are evaluated using SSD test data, the second one is online test where models interact with FSA-based user simulator, and the third one is human evaluation where models interact with humans.

The offline evaluation results of the three baseline models across all domains on SSD are summarised in Table 4. As we can see, all three models perform poorly, and nearly all the dialog success rates are lower than 50%. Remind that the success rate of UBAR on MultiWOZ is higher than 70%. Among them, GPT2 based models (SimpleTOD and UBAR) achieve relatively good performance on SSD owing to the efficacy of large pre-trained language models. Although SimpleTOD achieves the best results on all four domains. Nevertheless, SimpleTOD only reaches nearly 40% dialog success on SSD-PHONE and SSD-ID, 41.50% on SSD-NAME, and 36.58% on SSD-PLATE. Table 5 illustrates the results of online evaluations. The similar observations are concluded as those in offline evaluations. Even the most efficient SimpleTOD model achieves poor success rates.

From the detailed analysis of the results, we observe that one of the major factors affecting the performance is the difficulty of sub-slot locating, especially when updating a fragment of the sub-slot. In the phone number domain and ID number domain, the system should compare the updated fragment with the collected value to determine which fragment is similar to that one. As shown in Figure 5, the system is required to change “307” to “807”, but it wrongly updates “4307” to “807”. For the name slot, the system changes “陈” to “何”.

4.3 Performance of plug-in unit

Table 6 shows the performance of our knowledge plug-in unit on SSD-NAME. UBAR+ performs the best, with 23% improvement over UBAR and 6% improvement over SimpleTOD in terms of dialog succ. We claim that the knowledge plug-in unit enables the model to obtain relevant knowledge by querying the knowledge base, which is beneficial to complete slot value acquisition and response generation.

Further investigation is conducted through interaction between the model and the user simulator. Table 6 shows UBAR+ harvests a great improvement in name collecting, yielding an accuracy rate of 45.8%, which further proves the efficiency of knowledge-rich disambiguation. The same trend is
Table 6: Comparisons between UBAR+ and the SOTA models in both offline and online tests on the Chinese name domain.

| Model    | Offline Test | Online Test |
|----------|--------------|-------------|
|          | Joint acc    | Slot acc    | Dialog succ | Avg turn | Dialog succ |
| SimpleTOD| 79.22        | 91.24       | 51.50       | 4.79     | 15.80       |
| UBAR     | 63.58        | 82.58       | 34.40       | 4.41     | 11.50       |
| UBAR+    | **84.96**    | **93.12**   | **57.73**   | **4.60** | **45.80**   |

Table 7: Performance on human evaluation on Chinese name domain. App indicates the average appropriateness scores.

| Model    | Dialog succ | App | Diversity |
|----------|-------------|-----|-----------|
| UBAR     | 25.00       | 2.82| 3.10      |
| UBAR+    | **50.00**   | **2.89** | **3.96** |

4.4 Human Evaluation

Table 7: Performance on human evaluation on Chinese name domain. App indicates the average appropriateness scores.

For human evaluation, 10 postgraduates are recruited to evaluate UBAR+ and UBAR on Chinese name domain. During the interaction, the students randomly change the characters to those with similar pronunciations in the sentences. The same name and knowledge with errors are used on both models. At the end of the conversation, the evaluators are asked to check whether the dialog is successful. The postgraduates also score each system response to evaluate the appropriateness of the system response (Zhang et al., 2020a). The points range from 1 to 3, which respectively represent invalid, ok, and good. Another score on a Likert scale of 1-5 evaluates the diversity of the whole dialog. The results are shown in Table 7 and prove that UBAR+ yields a much higher dialog success rate.

5 Related Work

We can group the datasets for task-oriented dialog systems by whether the two parts involved in the dialogs are humans or machines: human-to-human (H2H), machine-to-machine (M2M) and human-to-machine (H2M) collecting methods. H2H corpora are derived by asking a human user to talk with a human agent. To mimic the conversations between human and machine, H2H datasets ubiquitously apply the Wizard-of-Oz approach (Hemphill et al., 1990; El Asri et al., 2017; Budzianowski et al., 2018; Zhu et al., 2020), which a human agent pretends as machine to talk to a human user and the human user believes the other side is a machine. However, it costs tremendous effort to construct such a H2H dataset. M2M datasets which are generated by simulated systems and simulated users take much less work to construct than H2H datasets with the same scale. However, the naturalness and diversity of M2M datasets are questioned (Peng et al., 2017; Shah et al., 2018; Rastogi et al., 2020; Dai et al., 2020). H2M (Raux et al., 2005; Williams et al., 2013; Henderson et al., 2014a,b; Kim et al., 2016) hires crowd workers to chat with a machine system and the conversations are more diverse and natural than M2M. We integrate the M2M and H2M approaches by boosting the generated M2M datasets through crowdsource rewriting to obtain more diverse and natural dialogs with less effort.

The datasets might be also grouped by the single-domain and the multi-domain. The early datasets are mostly single-domain. For example, ATIS (Hemphill et al., 1990), by M2M strategy, is a system to help people make air travel plans; a H2M corpus, Let’s Go Public (Raux et al., 2005), contains consultation dialogs of bus schedule information; two datasets for buying a movie ticket and reserving a restaurant table are collected by M2M (Shah et al., 2018). Single-domain systems generally fill slots within a single turn and thereby slot values are relatively independent. Recently, multi-domain datasets grab more attention. MultiWOZ (Budzianowski et al., 2018), one of the most popular datasets, consists of Wizard-of-Oz large-scale multi-domain conversations. A M2M dataset, SGD (Rastogi et al., 2020), generates multi-domain dialogs, guided by the predefined schema. Cross-WOZ (Zhu et al., 2020) states how slots in one domain relate to the following domains by reference. Nevertheless, none of the above datasets, with single domain or multiple domains, look into sub-slot cases as SSD does. In SSTOD, we have to not only locate the related previous sub-slots through complicated expressions, but also tile the pieces of value into a correct sequence without duplication, missing, and errors under the assistance of external knowledge.

6 Conclusions and Future Work

In this paper, we propose a sub-slot based task SSTOD which has not brought to the public. To help the exploration of the task, we build a textual dialog dataset SSD which covers four popular domains and contains natural noise brought by ASR module. SSD stems from the real human-to-human dialogs and can be utilized as a benchmark for slot
filling, dialog state tracking and dialog system that matches the real-world scenarios.

**Ethical Considerations**

The collection of our SSD dataset is consistent with the terms of use of any sources and the original authors’ intellectual property and privacy rights. The SSD dataset is collected with ALIDUTY\(^1\) platform, and each HIT requires up to 10 minutes to complete. The requested inputs are general language variations, speech voices, and no privacy-related information is collected during data collection. Each HIT was paid 0.1-0.2 USD for a single turn dialog data, which is higher than the minimum wage requirements in our area. The platform also hires professional reviewers to review all the collected data to ensure no ethical concerns e.g., toxic language and hate speech.

**Acknowledgements**

The authors would like to thank anonymous reviewers for their suggestions and comments, as well as professors for great inspiration. The authors are also grateful for all teammates for their productive and constructive advice.

**References**

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. journal of machine learning research, vol. 3, no.

Pawel Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Ilitgo Casanueva, Stefan Ultes, Osman Ramadán, 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.

Yinpei Dai, Hangyu Li, Yongbin Li, Jian Sun, Fei Huang, Luo Si, and Xiaodan Zhu. 2021. Preview, attend and review: Schema-aware curriculum learning for multi-domain dialogue state tracking. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 879–885, Online. Association for Computational Linguistics.

Yinpei Dai, Hangyu Li, Chengguang Tang, Yongbin Li, Jian Sun, and Xiaodan Zhu. 2020. Learning low-resource end-to-end goal-oriented dialog for fast and reliable system deployment. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 609–618, Online. Association for Computational Linguistics.

Layla El Asri, Hannes Schulz, Shikhar Sharma, Jeremie Zumer, Justin Harris, Emery Fine, Rahul Mehrotra, and Kaheer Suleman. 2017. Frames: a corpus for adding memory to goal-oriented dialogue systems. In Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue, pages 207–219, Saarbrücken, Germany. Association for Computational Linguistics.

Donghoon Ham, Jeong-Gwan Lee, Youngsoo Jang, and Kee-Eung Kim. 2020. End-to-end neural pipeline for goal-oriented dialogue systems using GPT-2. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 583–592, Online. Association for Computational Linguistics.

Wanwei He, Yinpei Dai, Yinhe Zheng, Yuchuan Wu, Zheng Cao, Dermot Liu, Peng Jiang, Min Yang, Fei Huang, Luo Si, et al. 2022. Galaxy: A generative pre-trained model for task-oriented dialog with semi-supervised learning and explicit policy injection. AAAI.

Charles T. Hemphill, John J. Godfrey, and George R. Doddington. 1990. The ATIS spoken language systems pilot corpus. In Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27, 1990.

Matthew Henderson, Blaise Thomson, and Jason D. Williams. 2014a. 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.

Matthew Henderson, Blaise Thomson, and Steve Young. 2014b. Word-based dialog state tracking with recurrent neural networks. In Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL), pages 292–299, Philadelphia, PA, U.S.A. Association for Computational Linguistics.

Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. A simple language model for task-oriented dialogue. In Advances in Neural Information Processing Systems, volume 33, pages 20179–20191. Curran Associates, Inc.

Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. Journal of documentation.
Seokhwan Kim, Luis Fernando D’Haro, Rafael E. Banchs, Jason Williams, and Matthew Henderson. 2016. The fourth dialog state tracking challenge. In Proceedings International Workshop on Spoken Dialogue Systems (IWSDS).

Xiujun Li, Yun-Nung Chen, Lihong Li, and Jianfeng Gao. 2017. End-to-end task-completion neural dialogue systems. CoRR, abs/1703.01008.

Ilya Loshchilov and Frank Hutter. 2018. Fixing weight decay regularization in adam.

Baolin Peng, Chunyuan Li, Jinchao Li, Shahin Shayandeh, Lars Liden, and Jianfeng Gao. 2020. SOLOIST: few-shot task-oriented dialog with A single pre-trained auto-regressive model. CoRR, abs/2005.05298.

Baolin Peng, Xiujun Li, Lihong Li, Jianfeng Gao, Asli Celikyilmaz, Sungjin Lee, and Kam-Fai Wong. 2017. Composite task-completion dialogue policy learning via hierarchical deep reinforcement learning. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 2231–2240, Copenhagen, Denmark. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2018. Language models are unsupervised multitask learners.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In AAAI, pages 8689–8696. AAAI Press.

Antoine Raux, Brian Langner, Dan Bohus, Alan W Black, and Maxine Eskenazi. 2005. Let’s go public! taking a spoken dialog system to the real world. In Ninth European conference on speech communication and technology.

Pararth Shah, Dilek Hakkani-Tür, Gokhan Tür, Abhinav Rastogi, Ankur Bapna, Neha Nayak, and Larry Heck. 2018. Building a conversational agent overnight with dialogue self-play.

Ching-Ho Tsai, N.J.C. Wang, P. Huang, and Jia-Lin Shen. 2005. Open vocabulary chinese name recognition with the help of character description and syllable spelling recognition. In Proceedings. (ICASSP ’05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005., volume 1, pages I/1037–I/1040 Vol. 1.

Nick Jui Chang Wang. 2007. An interactive open-vocabulary chinese name input system using syllable spelling and character description recognition modules for error correction. IEICE TRANSACTIONS on Information and Systems, 90(11):1796–1804.

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.

Liang Xu, Xuanwei Zhang, and Qianqian Dong. 2020. Cluecorpus2020: A large-scale chinese corpus for pre-training language model.

Yunyi Yang, Yunhao Li, and Xiaojun Quan. 2021. Ubar: Towards fully end-to-end task-oriented dialog systems with gpt-2.

Yichi Zhang, Zhijian Ou, and Zhou Yu. 2020a. Task-oriented dialog systems that consider multiple appropriate responses under the same context. In The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, pages 9604–9611. AAAI Press.

Zheng Zhang, Ryuichi Takanobu, Minlie Huang, and Xiaoyan Zhu. 2020b. Recent advances and challenges in task-oriented dialog system. CoRR, abs/2003.07490.

Qi Zhu, Kaili Huang, Zheng Zhang, Xiaoyan Zhu, and Minlie Huang. 2020. CrossWOZ: A large-scale Chinese cross-domain task-oriented dialogue dataset. Transactions of the Association for Computational Linguistics, 8:281–295.
A Knowledge

| char | knowledge                                      | explanation |
|------|-----------------------------------------------|-------------|
| 黄   | 草头黄 ('黄' has the radical '⺾')               |             |
| 林   | 双木林 ('林' has double '木')                  |             |
| 王   | 三横一竖王 (Three horizontal bars and one vertical bar 'wang') |             |
| 赢   | 亡口月贝凡 ('赢' is composed of '亡', '口', '月', '贝' and '凡') |             |
| 意   | 竖心旁加一个台湾的'台'的那个意 ('怡' is a combination of the radical '⺾' and '台' from Taiwan.) |             |

Figure 6: Some different types of knowledge on Chinese characters.

| char | knowledge                                      | explanation |
|------|-----------------------------------------------|-------------|
| 宝   | 宝宝 Baby                                     |             |
| 宝贵 | Precious                                      |             |
| 宝马 | BMW                                           |             |
| 宝箱 | Taobao                                        |             |
| 宝石 | Gemstone                                      |             |
| 宝藏 | Treasure                                      |             |
| 宝玉 | Precious jade                                 |             |
| 宝物 | Gems                                          |             |
| 宝箱 | Treasure Chest                                |             |
| 支付宝 | Alipay                                      |             |
| 宝盖头 | Chinese radical '⺾'                          |             |
| 小宝贝儿 | Little Baby |             |

Figure 7: Some pieces of knowledge about 宝.

B NLG Templates

| Domain | act | template, example and explanation |
|--------|-----|-----------------------------------|
| NAME   | inform | My surname is 'Wu', 'mouth' and 'fly'.'Wy'. My name is 'Wu', 'mouth' and 'fly'. |
| update |      | My name is 'Wu', 'mouth' and 'fly'. |

| PHONE | inform | You can take notes now, 188. |
| update |       | Last is 952, remember? |

Some NLG templates are presented in Figure 8. In the domain of name, Chinese name consists of surname "<sn>" and given name "<gn>". Each word in name has two kinds of knowledge, components "<cmpnt>" and words "<_words>", to distinguish different characters. In the phone domain, when generating one sentence using a template, 'X' is replaced by a sub-slot value to be informed and 'Y' is replaced by the noisy sub-slot value or which to be updated.
Taking the construction process of SSD-PHONE dataset as an example, the partial automata constructed by analyzing the H2H dialog is shown in Figure 9. When a system act is chosen, the user transfers state according to the user state and the system slot value and outputs the user act with probability. At the same time, in order to imitate various user expressions in the real world, one user act can correspond to various expressions as shown in Table 3.

Figure 9: Part of automata that generates the dialog of SSD-PHONE dataset and SSD-ID dataset.

Figure 10: Examples of dialogs of phone number collection and identity number collection.
| Domain | Dialog |
|--------|--------|
| NAME   | sys މ May I have your name?)
| user ៍ጱ៍ (Li Ling. “៍” as in “៍”).
| sys ҏ(Is “៍” the one “ᜎ” with a bottom which is the right part of “陵” as in “陵”?)
| user ੒ጱ̶(Yes, yes.)
| sys ੒ጱᦾ无情女生 (Is “圩” composed of “圩” and “女”?)
| user ੒ጱ҅圩 (No, nope, nope.)
| sys ፳ு (“圩” with a bottom which is the right part of “ጱ” as in “ጱ”.)
| user ੒ጱ̶ (“圩” as in “圩”)
| sys ፳ุฎ芷 (What should it be?)
| user ੒ጱ̶ (“圩” as in “圩”)
| sys ፳ு (Bye.)
| PLATE  | sys ੒ጱ̀፳ূ (Could you tell me your license plate number?)
| user ੒ጱ̀፳ূ (“GPSC”)
| sys ੒ጱሥ (The first digit is “ฤ” as in “ฤ”, right?)
| user ੒ጱ̶ (“ฤ” as in “ฤ”)
| sys ੒ጱօuls (The fifth digit is “ฤ”, not “ฤ”, “ฤ” as in “ฤ”)