DSTEA: Dialogue State Tracking with Entity Adaptive Pre-training

Yukyung Lee 1  Takyoung Kim 1  Hoonsang Yoon 1  Pilsung Kang 1  Junseong Bang 2  Misuk Kim 3

Abstract
Dialogue state tracking (DST) is a core submodule of a dialogue system, which aims to extract the appropriate belief state (domain-slot-value) from a system and user utterances. Most previous studies have attempted to improve performance by increasing the size of the pre-trained model or using additional features such as graph relations. In this study, we propose dialogue state tracking with entity adaptive pre-training (DSTEA), a system in which key entities in a sentence are more intensively trained by the encoder of the DST model. DSTEA extracts important entities from input dialogues in four ways, and then applies selective knowledge masking to train the model effectively. Although DSTEA conducts only pre-training without directly infusing additional knowledge to the DST model, it achieved better performance than the best-known benchmark models on MultiWOZ 2.0, 2.1, and 2.2. The effectiveness of DSTEA was verified through various comparative experiments with regard to the entity type and different adaptive settings.

1. Introduction
The main purpose of dialogue state tracking (DST) is to track the structured belief state (domain-slot-value) accurately within a dialogue. The performance of a dialogue system depends heavily on that of its DST system. Moreover, since the dialogue system mainly focused on the current dialogue turn, generating the correct multi-turn belief state is a challenging task (Kim et al., 2020).

To address this difficulty, recent studies have attempted to improve DST performance by applying a pre-trained model (Lin et al., 2020; Zhao et al., 2021) trained with a large number of unlabeled corpora, such as BART (Lewis et al., 2020) and T5 (Raffel et al., 2020), or by using a graph structure to inject additional information (Zhu et al., 2020; Wu et al., 2020). Alternatively, performance improvements were sought by conducting additional training (Mehri et al., 2020) using an open external dialogue dataset. However, when a large pre-trained model is used, an increase in inference time is inevitable, making this strategy difficult to use in a real-world dialogue system. The same issue arises when additional training data are used, or when a new graph structure is embedded.

To solve the limitations of previous DST models, the ideal strategy is to apply effective training with only a given dataset, rather than accepting the increased model complexity or training time caused by the use of additional data (Guu et al., 2020). To this end, we noticed that the dialogue utterance already contains key information, i.e., an entity that appears in one sentence, such as identifying people, places, and dates. Therefore, we propose DSTEA (Dialogue State Tracking with Entity Adaptive pre-training), a methodology in which the representation of the DST model trains entities in depth after extracting important information from a given utterance.

As shown in Figure 1, previous DST model used a BERT-based model trained by masking tokens selected at random, without considering the characteristics of such tokens. To improve upon this, we identified entities appearing in a dialogue and proposed the use of selective knowledge masking (SKM), which is a method for efficient pre-training by distinguishing the identified entities from relatively less important factors. This strategy means that DSTEA learns important knowledge at a higher masking rate because we assume that word and phrase entities contain indispensable knowledge and they are worth being trained more frequently than non-entity tokens. This can boost DST performance without increasing the parameter size of the model. Through this approach, the encoder, which plays a key role in the DST model, can learn the inductive bias suitable for dialogue state tracking as part of the additional pre-training process.

In this study, we conducted comparative experiments on various versions of the representative DST dataset, MultiWOZ (Budzianowski et al., 2018), to evaluate the performance of adaptive pre-training. Moreover, we verified DSTEA through an additional qualitative analysis of the
DSTEA: Dialogue State Tracking with Entity Adaptive Pre-training

Figure 1. Comparison between original and proposed masking strategies. We used selective knowledge for effective pre-training on the DST. We give the masking probability by distinguishing between knowledge selected as important information ($p_{\text{select}}$) or relatively insignificant information ($p_{\text{origin}}$).

In this study, previously proposed DST model achieved excellent performance through an improvement in the pre-trained encoder. We optimized the encoder for the dialogue domain using the proposed method based on SOM-DST, following the settings of Zhu et al. (2020).

2. Pre-training Methods

2.1. Adaptive Pre-training

Adaptive pre-training refers to a process in which a language model trained in the general domain is pre-trained to learn the knowledge suitable for a specific domain or task. Previous approaches reported enhanced performance for a specific domain or task by adding an adaptation phase between pre-training and fine-tuning (Gururangan et al., 2020; Beltagy et al., 2019).

In the area of DST, some studies, such as DialoGLUE (Mehri et al., 2020) and Zhao et al. (2021), have attempted to inject adaptive knowledge into their models. DialoGLUE performed additional pre-training to adapt the model to the conversation domain using various open dialogue datasets. During adaptive pre-training, DialoGLUE used the original masked language model (MLM). In the case of Zhao et al. (2021), various pre-training objectives were applied to T5, a pre-training model larger than BERT, and achieved good performance without any pre/post-processing. However, the use of a large amount of external dialogue data to improve the performance of DialoGLUE, or an encoder-decoder model that contains more knowledge in Zhao et al. (2021), has an intrinsic limitation imposed by the increased training or inference time.

Therefore, we focused on adaptive pre-training for entity recognition without significantly increasing the time complexity. In this process, external dialogue data are not used, and instead of large models such as BART or T5, ERNIE (Sun et al., 2020) was used with the same parameter size as BERT to maintain the training and inference time.

2.2. Masking Strategy

Since BERT first learned general-purpose knowledge through masked language modeling, Transformer-based language models have improved the training ability of such models through various masking strategies. Although BERT applies masking based on a single token, in subsequent studies, masking in units of tokens, words, and phrases was proposed to improve the language modeling performance (Sun et al., 2020).

ERNIE is a language model that contains entity information. It defines meaningful tokens, entities, and phrases as knowledge, and was the first model to perform continual learning of high-level knowledge during the pre-training process. SpanBERT (Joshi et al., 2020) learns a continuous
random span. Because the span boundary representation is learned without relying on individual tokens, it is more advanced than the original BERT. REALM (Guu et al., 2020) proposed salient-span masking to learn only the entity information for question answering. Pre-training was conducted after defining named-entity and date information as masking targets with a BERT-based tagger. Pegasus (Zhang et al., 2020b) proposed a gap sentence generation method to predict masked sentences based on the sentences that remain after masking in units of sentences from the input document. Approximately 30% of the sentences in the given documents could be masked.

In this study, we found that the knowledge definition of ERNIE and the entity masking of REALM are well suited to the nature of dialogues. Therefore, based on these techniques, we propose SKM to focus on the important entities.

3. Proposed Method

In this study, we propose DSTEA, which enhances pre-training for DST by focusing on important knowledge. DSTEA applies entity adaptive pre-training using system and user utterances to learn representations that are effective for dialogue state tracking. Subsequently, dialogue state tracking is performed using a well-trained encoder.

3.1. Overview

The overall architecture of the DSTEA is shown in Figure 2. The previous model applied DST using a pre-trained language model to conduct masked language modeling using a fixed random probability value without considering the characteristics of the token; this is the original masking strategy shown in Figure 1 (a). On the other hand, DSTEA assumes that the entity information appearing in the utterance is significant knowledge for DST; and applies entity adaptive pre-training using a SKM strategy shown in Figure 1 (b).

After applying this pre-training, DSTEA learns dialogue state tracking by utilizing an operation-based model, called the SOM-DST, which achieves good performance and consists of encoder and decoder structures, where the operation is the value of the slot-meta comprising the domain and slot. An operator can decide whether to create a value. Entity-adaptive-trained ERNIE is used as the encoder of the model proposed in this study, and the decoder uses a gated recurrent unit (GRU) network (Cho et al., 2014).

3.2. Entity Adaptive Pre-training

3.2.1. Why Entities?

The key to DST involves extracting information from the dialogue utterance between the system and the user and assigning it to an appropriate domain and slot. Most of the values that DST ultimately seeks pertain to an entity. Therefore, DSTEA attempts to improve performance by enriching the meaning of dialogue utterance through entity information.

3.2.2. Why ERNIE?

ERNIE is a framework for improving the performance of BERT, in which pre-training is conducted by injecting entity information into BERT as prior knowledge. The masking method in ERNIE differs from that of BERT in that it can mask entities or phrases rather than words. As such, ERNIE is a language model that learns using entity information, and is therefore suitable to the direction pursued by DSTEA. Thus, we determined that it would perform well for DST.

3.3. Entity Adaptive Pre-training Method

The encoder architecture is an essential part of the DST model. The purpose of the encoder model in the proposed method is to learn an inductive bias suitable for DST during
the pre-training process so that the representation of the pre-training model can be used to learn the dialogue information more accurately. The previous DST model used an encoder that trained the MLM with a random probability without considering the characteristics of the token. Using it without any modifications results in limited performance improvement, which we attempted to overcome by devising a pre-training method specialized for the DST.

The adaptive pre-training method proposed for DSTE:A is shown in Figure 3. Pre-training comprises three steps: entity construction, selective knowledge masking, and adaptive pre-training. After extracting an entity from the utterance, a higher masking rate was assigned by defining the information as important knowledge, and the original masking rate was assigned to the remaining tokens. In this study, training was conducted by considering both the word and phrase entities.

3.3.1. Entity Construction

One of the most important parts of entity adaptive pre-training is entity construction. In this study, the entities were collected in four ways. First, an entity was selected using ontology information. This is because an ontology is the most readily available form of information from the dataset, and it specifies the most important words. Second, after establishing a named entity recognition (NER) model, inference was conducted on the MultiWOZ dataset to collect the entities. An ERNIE-based BIO tagging model was used, and the model was trained using the CONLL 2003 dataset (Tjong Kim Sang & De Meulder, 2003). Third, entities were extracted using the spaCy (Honnibal & Montani, 2017) library. The spaCy entity recognizer extracts entities in span units, including entity types such as location, language, person, and product. Finally, entities were extracted using the flair library (Akbik et al., 2019). The flair entity tagger is a model that is trained based on CONLL 2003 and extracts entities in span units.

The entities extracted through these methods are a mixture of words and phrases. We attempted to learn these by distinguishing between word and phrase entities. The ontology was used during the pre-training process, but not during model training.

Word Entity First, each extracted entity was split into word units to compose a word entity. To prevent overfitting during this process, information about time and numbers was excluded from the entity. Because random times and numbers are used when constructing dialogue datasets (Budzianowski et al., 2018), the appearance of unseen information during a dialogue may prevent the DST model from responding correctly to unseen slots or values. Therefore, if we train entities regarding time and numbers with SKM, the biased model is highly likely to generate incorrect values; therefore, we did not take these values into account. Additionally, stopwords from the NLTK (Bird et al., 2009) library were used to exclude words that were not helpful for training. Moreover, filtering was conducted when the punctuation mark was extracted as an entity.

Phrase Entity To learn a phrase entity, cases that included an unknown token (i.e., [UNK]) in the phrase were excluded. Next, the phrase entities defined for each utterance were extracted in advance, and then pre-training was performed using randomly selected phrase entities. Even if a phrase entity included stopwords, filtering was not conducted for masking in span units, and information about numbers and times was excluded, as for word entities. However, phrase entities extracted by the NER model and flair were of poor quality; therefore, entities were extracted using only the ontology and spaCy.

3.3.2. Selective Knowledge Masking Strategy

SKM is a method for learning important knowledge after selecting essential information from user and system utterances, to inject an inductive bias suitable for DST. The pre-training model was trained as an appropriate model for DST by masking and predicting entities that can help track the dialogue state, rather than arbitrary mask tokens. SKM is shown in step 2 of Figure 3. The one-turn utterance to-
Table 1. JGA on the test sets of MultiWOZ 2.0, 2.1, and 2.2. * indicates a model that was re-implemented. We also describe the pre-trained model used by each methodology. Both BERT and ERNIE are the results of the experimentation with the base model. (Parameter size: BERT - 110M, ERNIE - 110M, BART (base) - 140M, T5 (base) - 220M)

| Model                  | PLM    | MultiWOZ 2.0 | MultiWOZ 2.1 | MultiWOZ 2.2 |
|------------------------|--------|--------------|--------------|--------------|
| SUMBT* (Lee et al., 2019) | BERT   | 42.07        | 46.99        | -            |
| TRADE* (Wu et al., 2019)  | -      | 47.80        | 49.38        | 49.97        |
| DSDST                  | BERT   | 52.24        | 51.21        | -            |
| GCDST                  | -      | 50.68        | 46.09        | -            |
| SOM-DST* (Kim et al., 2020) | BERT   | 51.38        | 52.75        | 53.70        |
| SOM-DST + Schema Graph (Zhu et al., 2020) | BERT | 51.57 | 52.88 | - |
| SST-1 (Chen et al., 2020) | BERT   | 47.59        | 52.55        | -            |
| MinTL Lin et al., 2020 | T5     | 51.24        | 52.52        | -            |
| MinTL Lin et al., 2020 | BART   | 52.07        | 53.62        | -            |
| Pegasus (Zhao et al., 2021) | T5     | -            | 54.40        | -            |
| **DSTEA (ours)**       | ERNIE  | 54.11        | 55.03        | 55.23        |

Table 2. JGA, SA and RSA scores by domain for MultiWOZ 2.0, 2.1, and 2.2. The performance was compared with the re-implemented TRADE (Wu et al., 2019), SUMBT (Lee et al., 2019), and SOM-DST (Kim et al., 2020) models. (* indicates the re-implemented model.)

| Domain | Model                  | MultiWOZ 2.0 | MultiWOZ 2.1 | MultiWOZ 2.2 |
|--------|------------------------|--------------|--------------|--------------|
| **Attraction** | **JGA** | **SA** | **RSA** | **JGA** | **SA** | **RSA** | **JGA** | **SA** | **RSA** |
| SUMBT* | 59.21                 | 98.42        | 80.29        | 63.19        | 98.59        | 81.95        | 64.25        | 98.61        | 81.91        |
| TRADE* | 59.22                 | 98.42        | 79.32        | 63.84        | 98.60        | 81.34        | 68.02        | 98.78        | 84.48        |
| SOM-DST* | + Schema Graph (Zhu et al., 2020) | **BART** | -            | -            | -            | -            | -            | -            | -            |
| DSDST (Ours) | 69.21                 | 98.68        | 83.21        | 68.11        | 98.78        | 84.21        | -            | -            | -            |
| **Hotel** | **JGA** | **SA** | **RSA** | **JGA** | **SA** | **RSA** | **JGA** | **SA** | **RSA** |
| SUMBT* | 53.29                 | 96.19        | 77.84        | 44.05        | 96.95        | 82.61        | 48.55        | 97.27        | 83.52        |
| TRADE* | 47.08                 | 97.20        | 83.19        | 49.86        | 97.32        | 83.92        | 43.33        | 97.04        | 83.55        |
| SOM-DST* | + Schema Graph (Zhu et al., 2020) | **BART** | -            | -            | -            | -            | -            | -            | -            |
| DSDST (Ours) | 48.76                 | 97.26        | 84.69        | 48.61        | 97.33        | 84.93        | 49.23        | 97.37        | 85.24        |
| **Restaurant** | **JGA** | **SA** | **RSA** | **JGA** | **SA** | **RSA** | **JGA** | **SA** | **RSA** |
| SUMBT* | 57.95                 | 98.03        | 85.10        | 58.14        | 98.16        | 86.83        | -            | -            | -            |
| TRADE* | 61.59                 | 98.38        | 88.34        | 58.59        | 98.24        | 87.01        | 59.25        | 98.24        | 86.93        |
| SOM-DST* | + Schema Graph (Zhu et al., 2020) | **BART** | -            | -            | -            | -            | -            | -            | -            |
| DSDST (Ours) | 67.45                 | 98.66        | 90.36        | 69.88        | 98.70        | 91.07        | 68.67        | 98.67        | 90.46        |
| **Taxi** | **JGA** | **SA** | **RSA** | **JGA** | **SA** | **RSA** | **JGA** | **SA** | **RSA** |
| SUMBT* | 25.08                 | 95.51        | 52.95        | 31.96        | 95.88        | 56.18        | -            | -            | -            |
| TRADE* | 36.60                 | 96.64        | 64.27        | 33.18        | 96.59        | 63.16        | 36.29        | 96.65        | 64.02        |
| SOM-DST* | + Schema Graph (Zhu et al., 2020) | **BART** | -            | -            | -            | -            | -            | -            | -            |
| DSDST (Ours) | 61.21                 | 98.14        | 80.68        | 58.57        | 98.06        | 80.48        | 56.23        | 97.81        | 77.41        |
| **Train** | **JGA** | **SA** | **RSA** | **JGA** | **SA** | **RSA** | **JGA** | **SA** | **RSA** |
| SUMBT* | 57.68                 | 97.85        | 84.99        | 68.46        | 98.39        | 90.27        | -            | -            | -            |
| TRADE* | 66.90                 | 98.52        | 89.88        | 67.31        | 98.56        | 90.38        | 68.40        | 98.62        | 90.72        |
| SOM-DST* | + Schema Graph (Zhu et al., 2020) | **BART** | -            | -            | -            | -            | -            | -            | -            |
| DSDST (Ours) | 72.19                 | 98.74        | 91.88        | 73.88        | 98.81        | 92.32        | 73.65        | 98.80        | 92.28        |

We conducted experiments using MultiWOZ 2.0, 2.1, and 2.2 (Budzianowski et al., 2018; Eric et al., 2019; Zang et al., 2020), which are the most widely used datasets for DST. Similarly, as in previous DST model, we conducted an experiment using five domains (restaurant, train, hotel, taxi, and attraction). Most of the preprocessing procedures followed the script provided in TRADE (Wu et al., 2019).

4. Experimental Results

4.1. Datasets

4.2. Experimental Settings

We compared the performance of DSTEA with that of a number of previous models. We re-implemented those mod-
els for which either performance on the MultiWOZ datasets has not been published, or we were unable to reproduce their performance with the published code. In the case of SUMBT in MultiWOZ 2.2, the performance fluctuated greatly due to ontology update issues, so the experiment of MultiWOZ 2.2 was not reported. DSTEA only improves encoder models using dialogue utterance during adaptive training. Therefore, we conducted a performance comparison with models that include an encoder and do not use additional labels. For this reason, some well-known models were not considered in our experiments. First, Trippy (Heck et al., 2020) did not only use the belief state but also used an additional dialogue action label to improve the DST performance. In addition, SimpleTOD (Hosseini-Asl et al., 2020) is a model composed of only a decoder. Since DSTEA is a model that performed adaptive training based on an encoder, including a model with only a decoder in the benchmark models is not appropriate.

We trained the ‘pre-trained ERNIE-2.0’ on dialogue and used the huggingface transformers (Wolf et al., 2020), with ‘nguyoung/ernie-2.0-en’ as the ERNIE model. During the experiment, the masking probability \( \text{prob}_{\text{origin}} \) was set to 0.2, and \( \text{prob}_{\text{select}} \) was set to 0.4. Detailed parameters for adaptive training and DST training can be found in Appendix A.1.

### 4.3. Joint Goal Accuracy

We compared and analyzed the performance of DSTEA using joint goal accuracy (JGA), an evaluation metric that verifies whether the predicted belief states exactly match the gold label. Table 1 reports JGA values obtained with DSTEA and the baseline models for MultiWOZ 2.0, 2.1, and 2.2.

DSTEA achieved better performance than previous BERT-based DST model and also outperformed CSFN-DST and SOM-DST + Schema Graph, in which schema graph information is injected into SOM-DST. Furthermore, DSTEA achieved higher accuracy than MinTL using large pre-trained models such as T5 and BART.

In particular, through the performance difference between SOM-DST and DSTEA, we demonstrated experimentally that the SKM strategy configures the encoder as ERNIE in the pre-training process and masks information selectively. We also confirmed that more important entities are influential. DSTEA demonstrated this by outperforming the benchmark model for three datasets, with the only exception being when pre-training was conducted for MultiWOZ 2.2 on T5 based on the Pegasus pre-training objective. Although the number of parameters of the model proposed in Zhao et al. (2021) is approximately twice that of our model, DSTEA outperforms it for MultiWOZ 2.1. These results show that solving the DST task using the large pre-trained model is not optimal for performance enhancement and that training an inductive bias appropriate for the task helps improve performance.

### 4.4. Domain Results

For each dataset, domain-specific performance comparisons between the benchmark models and DSTEA are presented in Table 2. In addition to JGA, the performance was evaluated using slot accuracy (SA) and relative slot accuracy (RSA) (Kim et al., 2022). SA is an evaluation metric that identifies the accuracy of slots among the predicted dialogue states. RSA is a recently proposed metric that complements JGA and SA. In contrast to SA, RSA only focuses on the gold reference or predicted slots of current dialogue, rather than all pre-defined slots in SA, it has a better discriminative power for candidate models. The corresponding result was recorded as the performance of the experiment at the same random seed. The results show that DSTEA achieved the best performance in all metrics except Hotel in MultiWOZ 2.1 and Taxi in MultiWOZ 2.2.

### 4.5. Effectiveness of Adaptive Pre-training

Table 3 shows how the performance is affected by the adaptive pre-training setting. To demonstrate the effectiveness of SKM, the performance of the pre-trained model and SKM as
Figure 4. Domain, slot-meta, value mismatch between DST model (SUMBT, TRADE, SOM-DST, DSTEA) and ground truth in MultiWOZ 2.1 (Total turn 7368)

proposed in this study were compared by changing the masking probability of the original MLM. SOM-DST (ERNIE) indicates that the encoder of SOM-DST is changed to ERNIE, and DSTEA (ERNIE) + Random Masking \((P_{\text{origin}} = \alpha)\) indicates a case in which random masking probability \(\alpha\) is used during adaptive pre-training. The experiment involved adjusting the masking probability without considering the characteristics of the token. The results show that performance is improved by applying ERNIE, a language model learned by focusing the encoder on the entity. Moreover, when the masking probability was set to 0.2, performance was lower than that of SOM-DST (ERNIE); however, DSTEA outperformed SOM-DST (ERNIE) with a masking probability of 0.4. However, the best overall results were achieved with DSTEA (ERNIE) + Selective Masking \((P_{\text{select}} = \beta, P_{\text{origin}} = \alpha)\), which is the proposed method with SKM. Consequently, the experiment confirms that the proposed SKM method is more effective than simply increasing the masking probability.

Figure 5 shows the performance of DSTEA according to the change of SKM rate. The red dashed line represents the pre-training model that does not consider entities, which can be understood as the lower bound performance. The blue line indicates the performance of DSTEA according to each \(P_{\text{select}}\) ratio. SKM clearly enhanced the DST performance irrespective of \(P_{\text{select}}\) ratio. Although the best JGA was reported when \(P_{\text{select}} = 0.4\), the worst case still yielded a significantly enhanced JGA compared to that without SKM.

4.6. Effectiveness of Entity Types

We confirmed in the previous section that entities provide important information regarding the DST task, and compared and analyzed the performance of DSTEA for various entity types, as shown in Table 4. We extracted the entities through four modules, and both word and phrase units were considered. First, results show that an ontology is best for individual modules at the word level, whereas the performance is poor when word entities are extracted from flair only. However, when information of all entities was combined, better performance was achieved than when using individual modules. Phrase level entities also improve performance when training by combining the phrase entities. Finally, when both words and phrases were mixed, the best JGA of 55.03 was achieved. This is key knowledge for both word and phrase entities for DST, and it can be confirmed that configuring an appropriate entity set significantly improves performance.

4.7. Qualitative Result

We verified the superiority of our model through the qualitative results. Finally, the DST model generates prediction values consisting of domain-slot values for each turn. We compared the degree of mismatch ratio between the benchmark models and DSTEA with respect to the domain, slot-meta, and value. Figure 4 shows that DSTEA effectively reduces mismatching compared to the benchmark models. This means that the domain and slot information of the current utterance turn can be understood accurately, and a value suitable for each slot can be generated. These results suggest that the model trained using DSTEA can learn the appropriate inductive bias and improve its performance.

Table 5 shows a sample output for each model. The domain mismatch example shows that DSTEA can accurately classify domains in a dataset for a multi-domain dialogue. In addition, the slot-meta and value mismatch example confirms that DSTEA can match an appropriate slot corresponding to the domain and generate a relevant value.

5. Conclusion

In this study, we propose DSTEA, which applies entity adaptive pre-training by focusing on entities and essential
Table 5. Sample outputs of domain, slot-meta, and value mismatch in MultiWOZ 2.1 (test set). Red indicates a mismatch.

| SUMBT           | TRADE                        | SOM-DST                      | DSTEA          |
|-----------------|------------------------------|------------------------------|----------------|
| **Domain mismatch** (PMUL4648 - turn 2) |
| "hotel-name-nusha" | "attraction-name-nusha", "restaurant-name-nandos" | "restaurant-name-nusha", "attraction-name-nusha", |
|                  |                              |                              |                |
| **Slot-meta mismatch** (MUL0148 - turn 2) |
| "hotel-pricerange-expensive" | "hotel-pricerange-expensive", "hotel-type-hotel" | "hotel-pricerange-expensive", "hotel-type-hotel", |
|                  |                              |                              |                |
| **Value mismatch** (PMUL1106 - turn 2) |
| "train-departure-none", "train-departure-stansted airport", | "train-departure-stan", "train-departure-cambridge" | "train-departure-stan", "train-departure-cambridge", |

knowledge in dialogue state tracking. DSTEA uses ERNIE, a model specialized for entities. In addition, SKM was proposed to learn word and phrase entities effectively to focus on important information. Through this, DSTEA outperforms the benchmark models by learning an inductive bias suitable for dialogue state tracking during pre-training, and generating a suitable representation. DSTEA achieved satisfactory performance for MultiWOZ 2.0, 2.1, and 2.2 without using a large pre-trained model or directly injecting additional information into the model. We expect that this pre-training method can be used effectively for both dialogue state tracking under insufficient input data and entity-related tasks.

References

Akbik, A., Bergmann, T., Blythe, D., Rasul, K., Schweter, S., and Vollgraf, R. Flair: An easy-to-use framework for state-of-the-art nlp. In NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations), pp. 54–59, 2019.

Beltagy, I., Lo, K., and Cohan, A. SciBERT: A pretrained language model for scientific text. 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), pp. 3615–3620, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1371.

Bird, S., Klein, E., and Loper, E. Natural language processing with Python: analyzing text with the natural language toolkit. “O’Reilly Media, Inc.”, 2009.

Budzianowski, P., Wen, T.-H., Tseng, B.-H., Casanueva, I., Ultes, S., Ramadan, O., and Gašić, M. 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, pp. 5016–5026, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1547.

Chen, L., Lv, B., Wang, C., Zhu, S., Tan, B., and Yu, K. Schema-guided multi-domain dialogue state tracking with graph attention neural networks. Proceedings of the AAAI Conference on Artificial Intelligence, 34(05): 7521–7528, Apr. 2020. doi: 10.1609/aaai.v34i05.6250.

Cho, K., van Merriënboer, B., Bahdanau, D., and Bengio, Y. On the properties of neural machine translation: Encoder-decoder approaches. In Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation, pp. 103–111, Doha, Qatar, October 2014. Association for Computational Linguistics. doi: 10.3115/v1/W14-4012.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. 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), pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423.

Eric, M., Goel, R., Paul, S., Sethi, A., Agarwal, S., Gao, S., and Hakkani-Tur, D. Multiwoz 2.1: Multi-domain dialogue state corrections and state tracking baselines. arXiv preprint arXiv:1907.01669, 2019.

Gururangan, S., Marasović, A., Swayamdipta, S., Lo, K., Beltagy, I., Downey, D., and Smith, N. A. Don’t stop pretraining: Adapt language models to domains and tasks. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 8342–8360, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.740.
DSTEA: Dialogue State Tracking with Entity Adaptive Pre-training

Guu, K., Lee, K., Tung, Z., Pasupat, P., and Chang, M.-W. Realm: Retrieval-augmented language model pre-training. ArXiv, abs/2002.08909, 2020.

Heck, M., van Niekerk, C., Lubis, N., Geishauser, C., Lin, H.-C., Moresi, M., and Gašić, M. TripPy: A triple copy strategy for value independent neural dialo state tracking. In Proceedings of the 21st Annual Meeting of the Special Interest Group on Discourse and Dialogue, pp. 35–44, 1st virtual meeting, July 2020. Association for Computational Linguistics.

Honnibal, M. and Montani, I. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear, 2017.

Hosseini-Asl, E., McCann, B., Wu, C.-S., Yavuz, S., and Socher, R. A simple language model for task-oriented dialogue. arXiv preprint arXiv:2005.00796, 2020.

Joshi, M., Chen, D., Liu, Y., Weld, D. S., Zettlemoyer, L., and Levy, O. SpanBERT: Improving pre-training by representing and predicting spans. Transactions of the Association for Computational Linguistics, 8:64–77, 2020.

Kim, S., Yang, S., Kim, G., and Lee, S.-W. Efficient dialogue state tracking by selectively overwriting memory. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 567–582, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.53.

Kim, T., Yoon, H., Lee, Y., Kang, P., and Kim, M. Mismatch between multi-turn dialogue and its evaluation metric in dialogue state tracking, 2022.

Lee, H., Lee, J., and Kim, T.-Y. SUMBT: Slot-utterance matching for universal and scalable belief tracking. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 5478–5483, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1546.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 7871–7880, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.703.

Lin, Z., Madotto, A., Winata, G. I., and Fung, P. MinTL: Minimalist transfer learning for task-oriented dialogue systems. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 3391–3405, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.273.

Mehri, S., Eric, M., and Hakkani-Tur, D. Dialogue: A natural language understanding benchmark for task-oriented dialogue. ArXiv, abs/2009.13570, 2020.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research, 21(140):1–67, 2020.

Sun, Y., Wang, S., Li, Y., Feng, S., Tian, H., Wu, H., and Wang, H. Ernie 2.0: A continual pre-training framework for language understanding. Proceedings of the AAAI Conference on Artificial Intelligence, 34(05):8968–8975, Apr. 2020. doi: 10.1609/aaai.v34i05.6428.

Tjong Kim Sang, E. F. and De Meulder, F. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, pp. 142–147, 2003.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), Advances in Neural Information Processing Systems, volume 30. Curran Associates, Inc., 2017.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. Transformers: State-of-the-art natural language processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp. 38–45, Online, October 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-demos.6.

Wu, C.-S., Madotto, A., Hosseini-Asl, E., Xiong, C., Socher, R., and Fung, P. Transferable multi-domain state generator for task-oriented dialogue systems. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 808–819, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1078.

Wu, P., Zou, B., Jiang, R., and Aw, A. GCDST: A graph-based and copy-augmented multi-domain dialogue state tracking. In Findings of the Association for Computational Linguistics: EMNLP 2020, pp. 1063–1073, Online,
November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.95.

Zang, X., Rastogi, A., Sunkara, S., Gupta, R., Zhang, J., and Chen, J. 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, ACL 2020, pp. 109–117, 2020.

Zhang, J., Hashimoto, K., Wu, C.-S., Wang, Y., Yu, P., Socher, R., and Xiong, C. Find or classify? dual strategy for slot-value predictions on multi-domain dialog state tracking. In Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics, pp. 154–167, Barcelona, Spain (Online), December 2020a. Association for Computational Linguistics.

Zhang, J., Zhao, Y., Saleh, M., and Liu, P. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In III, H. D. and Singh, A. (eds.), Proceedings of the 37th International Conference on Machine Learning, volume 119 of Proceedings of Machine Learning Research, pp. 11328–11339. PMLR, 13–18 Jul 2020b.

Zhao, J., Mahdieh, M., Zhang, Y., Cao, Y., and Wu, Y. Effective sequence-to-sequence dialogue state tracking. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 7486–7493, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.593.

Zhu, S., Li, J., Chen, L., and Yu, K. Efficient context and schema fusion networks for multi-domain dialogue state tracking. In Findings of the Association for Computational Linguistics: EMNLP 2020, pp. 766–781, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.68.
DSTEA: Dialogue State Tracking with Entity Adaptive Pre-training

A. Appendix

A.1. Training details

Our models are built with the huggingface transformers. And we trained our model with NVIDIA GeForce RTX 3090.

| Hyperparameter     | Assignment |
|--------------------|------------|
| number of epoch    | 10         |
| batch size         | 16         |
| learning rate      | 6e-5       |
| optimizer          | AdamW      |
| weight decay       | 0.01       |
| warmup ratio       | 0.06       |
| selective masking ratio | 0.4     |
| original masking ratio | 0.2     |

Table 6. Hyperparameters for entity adaptive pre-training

| Hyperparameter     | Assignment |
|--------------------|------------|
| number of epochs   | 30         |
| batch size         | 64         |
| encoder learning rate | 4e-5    |
| decoder learning rate | 2e-4    |
| drop out           | 0.1        |
| warm up ratio      | 0.1        |
| max sequence length | 256     |

Table 7. Hyperparameters for DSTEA

A.2. Data statistics

| domain    | MultiWOZ 2.0, 2.1, 2.2 |
|-----------|------------------------|
|           | train  | dev   | test  |
| attraction| 2717   | 401   | 395   |
| hotel     | 3381   | 416   | 394   |
| restaurant| 3813   | 438   | 437   |
| taxi      | 1654   | 207   | 195   |
| train     | 3103   | 484   | 494   |

Table 8. Data statistics on MultiWOZ 2.0, 2.1 and 2.2

A.3. Entity extraction details

We use spaCy, flair library to extract entities. The settings we used are below.

Table 9. Settings for spaCy entity extraction

| Setting     | Assignment   |
|-------------|--------------|
| model       | "en-core-web-sm" |
| version     | 3.0.2        |

Table 10. Settings for flair entity extraction

| Setting      | Assignment          |
|--------------|---------------------|
| model        | "flair/her-english-large" |
| version      | 0.10                |