Preview, Attend and Review: Schema-Aware Curriculum Learning for Multi-Domain Dialog State Tracking

Yinpei Dai[†], Hangyu Li[†], Yongbin Li[†][*], Jian Sun[†], Fei Huang[†], Luo Si[†], Xiaodan Zhu[‡]

[†]Alibaba Group

[‡]Ingenuity Labs Research Institute & ECE, Queen's University

{yinpei.dyp,hangyu.lhy,shuide.lyb}@alibaba-inc.com
{jian.sun,f.huang,luo.si}@alibaba-inc.com, zhu2048@gmail.com

Abstract

Existing dialog state tracking (DST) models are trained with dialog data in a random order, neglecting rich structural information in a dataset. In this paper, we propose to use curriculum learning (CL) to better leverage both the curriculum structure and schema structure for task-oriented dialogs. Specifically, we propose a model-agnostic framework called **S**chema-**a**ware **C**urriculum **L**earning for Dia**log** State Tracking (SaCLog), which consists of a preview module that pre-trains a DST model with schema information, a curriculum module that optimizes the model with CL, and a review module that augments mispredicted data to reinforce the CL training. We show that our proposed approach improves DST performance over both a transformer-based and RNN-based DST model (TripPy and TRADE) and achieves new state-of-the-art results on WOZ2.0 and MultiWOZ2.1.

1 Introduction

Dialog state tracking (DST) extracts users' goals in task-oriented dialog systems, where dialog states are often represented in terms of a set of slot-value pairs (Williams et al., 2016; Eric et al., 2020). Due to the language variety of multi-turn dialogs, the concepts of slots and values are often indirectly expressed in the conversation (such as co-references, ellipsis, and diverse appearances), which are a major bottleneck for improving DST performance (Gao et al., 2019; Hu et al., 2020). Many existing DST methods have focused on designing better model architectures to tackle the problems (Dai et al., 2018; Wu et al., 2019; Kim et al., 2020), but still neglect the full exploitation of two important aspects of structural information.

The first is *curriculum structure* in a dataset. Such a structure relies on a measure of the difficulty of examples, which can be used to guide the
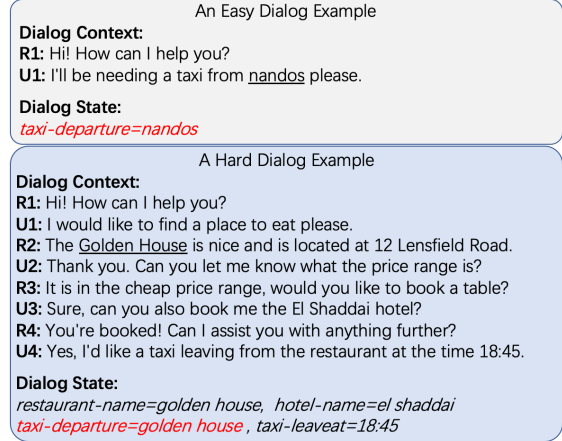
---

*Corresponding author

Figure 1: An easy and a hard dialog example for DST.

model training in an easy-to-hard manner, imitating the meaningful learning order in human curricula. This paradigm is called curriculum learning (CL) (Bengio et al., 2009) and has been shown useful in various other problems (Wang et al., 2021). DST training examples also vary greatly in their difficulty levels. As shown in Figure 1, for the same slot *'taxi-departure'*, a user can either inform its value *'nandos'* explicitly in a simple utterance or convey her intention implicitly via multi-round interactions, requiring a complex inference process to find the value *'golden house'* referred by the slot *'restaurant-name'*. However, CL has been rarely studied in DST, and models are often trained with dialog data in a random order.

In addition, *schema structure* is prominent in multi-domain task-oriented dialogs. A schema is specified by a collection of all possible slots and their values, which describes semantic relations among them. Some previous work utilized the structure via an extra schema graph in a regular training process (Chen et al., 2020; Zhu et al., 2020; Wu et al., 2020). We propose to incorporate schema information into CL through a pre-curriculum process, in which a DST model can be pre-trained with schema-related objectives to prepare for upcom-

ing DST examples. To reinforce the CL training, we can also expand those examples with frequent mispredictions during CL based upon the schema, enabling the model to accumulate more experience and perform better on similar cases.

Built on these motivations, we propose a novel framework named as **S**chema-**a**ware **C**urriculum **L**earning for Dia**log** State Tracking (SaCLog), which consists of three components: 1) a **preview module** that pre-trains the base part of a DST model (e.g., BERT and RNN) with objectives capturing the connections between the schema and dialog contexts, 2) a **curriculum module** that organizes training data from easy to hard and optimizes the model with CL, and 3) a **review module** which leverages schema-based data augmentation to extend mispredicted data to boost the CL training process further. The proposed approach is model-agnostic, in the sense that it can be incorporated into different DST models. To the best of our knowledge, this is the first attempt to apply CL to the DST task. We show that our proposed approach improves DST performance over both a transformer-based and RNN-based DST model (TripPy and TRADE) and achieves new state-of-the-art results on WOZ2.0 and MultiWOZ2.1.

2 Problem Formulation

We denote a dialog context containing t turns as $X_t = \{(R_1, U_1), ..., (R_t, U_t)\}$, where R_i and U_i represent system and user utterance at the i-th turn respectively. DST is tasked to extract turn-level or discourse-level dialog states in the form of a set of slot-value pairs given X_t. A turn-level dialog state $Y_t = \{(s, v_t), s \in \mathcal{S}\}$ is the slot-value pairs extracted only from (R_t, U_t) at current turn t, where $\mathcal{S}$ is a predefined set of slot s in the schema and v_t is the corresponding value[1] of the slot s. A discourse-level dialog state Z_t is the accumulation of L_t, representing all slot-value pairs that have been expressed over the course of the dialog until the t-th turn. We denote a dialog data for DST as $d_t = \{X_t, Y_t, Z_t\}$ and the training dataset as $\mathcal{D}$.

3 Schema-Aware Curriculum Learning

In this section, we first introduce the core curriculum module about how to apply the basic CL to the DST task; we then describe the preview and review module, which exploit the schema structure

to facilitate the CL training process. The overall framework of SaCLog is shown in Figure 2.

3.1 Curriculum Learning for DST

We propose curriculum learning for DST and design two sub-modules: a **difficulty scorer** that measures the difficulty level of a dialog example with respect to a DST model, as well as a **training scheduler** module that arranges the scored data as a sequence of easy-to-hard training stages.

3.1.1 The Difficulty Scorer

As a dialog example could be intuitively *complex* for humans or inherently *difficult* for neural networks (NNs), both model-based and rule-based scores should be considered. We propose to use a hybrid scoring function that combines the advantages of model predictions and rules.

For model-based difficulty, we predict scores in a cross-validation-like manner. We divide $\mathcal{D}$ into K equal-sized subsets, where $K-1$ subsets are used to train a DST model to predict the remaining one. This process is repeated K times until every subset is predicted. The score $r_t^{mod} \in [0, 1]$ is computed based on the average accuracy of all mentioned slots (whose values are not *none*) in Y_t for each d_t. In our experiment, we train six models with the same architecture and different initialization seeds to obtain the mean value $\bar{r}_t^{mod}$ of model scores.

For rule-based difficulty, we consider 4 factors to fuse human prior knowledge about DST into our curriculum design: 1) current dialog turn number t; 2) the total token number of (R_t, U_t); 3) the number of mentioned name entities like 'hotel names' in Z_t; 4) the number of newly added or changed slots in Y_t. We set the maximum values of above factors as 7/50/4/6 respectively, and normalize all factors into $r_t^{rul,i} \in [0, 1]$, where i indicates the i-th factor.

Finally, the hybrid difficulty score is calculated jointly as $r_t^{hyb} = \alpha_0 \bar{r}_t^{mod} + \sum_{i=1}^{4} \alpha_i r_t^{rul,i}$, where $r^{hyb} \in [0, 1]$ and $\sum_{i=0}^{4} \alpha_i = 1$.

3.1.2 The Training Scheduler

We adopt a widely used strategy called *baby step* (Spitkovsky et al., 2010) to organize the scored data for CL. Specifically, we divide the score uniformly into N intervals and distribute the sorted data into N buckets accordingly. The optimization starts from the easiest bucket as the initial training stage. After reaching a fixed number of maximum epochs or convergence, the next bucket is merged

---

[1]Each s contains two special values, *none* and *dontcare*, indicating s has no values and can take any values respectively.
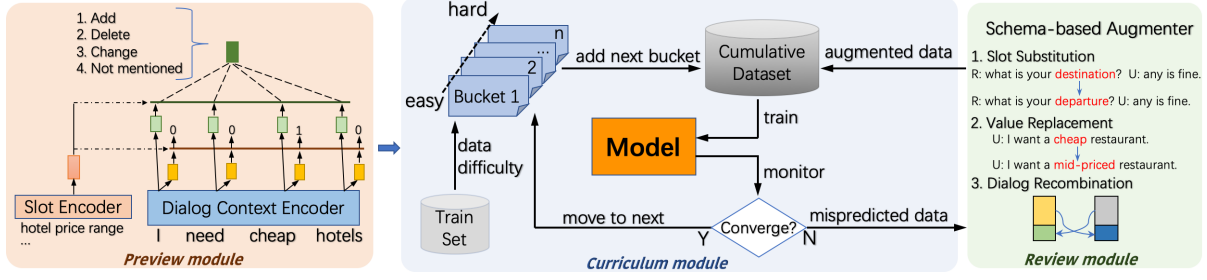
Figure 2: An overview of the SaCLog training procedures.

into the current training subset and shuffled for the next training stage. In our experiment, we set the maximum number of epochs as 3, and treat as the convergence if the training loss ceases to decrease and the loss value is within a threshold 15 for 100 steps. As the subset accumulates until all buckets are aggregated, we then continue to train the model for several extra epochs.

3.2 The Preview Module

In human learning, previewing learning materials helps develop an overall picture of what will be covering and can bring benefits to the learning process. In our task here, we propose new pre-training objectives to learn structural inductive bias of the schema structure. Specifically, our *preview* module contains a slot encoder to compute a slot embedding e_s for each input slot s, and a dialog context encoder to extract the hidden states of X_t as $E_t = [e_t^1, e_t^2, ...]$, then we have:

$$B_t^s = [\phi_1^{sig}(e_s \oplus e_t^1), \phi_1^{sig}(e_s \oplus e_t^2), ...]$$
$$c_t^s = \phi_4^{sft}(e_s \oplus \text{Att}(e_s, E_t))$$
(1)

where $\text{Att}(k, V)$ is the attention function using the vector k to query the vector sequence V to get a context vector and $\oplus$ the vector concatenation. $\phi_d^{sig}(\cdot)$ and $\phi_d^{sft}(\cdot)$ denote an FNN with one hidden layer having the same size as input layer, where the output layer is of size d, and is sigmoid and softmax respectively. B_t^s is a binary sequence indicates which span of X_t belongs to the value of s, while c_t^s is the classification logits indicates whether s is added, deleted, changed, or not mentioned in Y_t.

Therefore, for each slot s, we have a binary sequence loss L_{seq} and a classification loss L_{cls} to optimize. Such pre-training objectives help the encoders understand how a slot is roughly operated in the current dialog context and connected with all possible tokens regarding its values in the schema. The dialog context encoder is used for the parameter initialization of the base part of a DST model. The pre-trained corpus is constructed from

MultiWOZ2.1 dialogs (Eric et al., 2020) and the off-the-shelf synthesized dialogs (Campagna et al., 2020), which contains 337,346 dialog data in total.

We also leverage the language modelling (LM) loss as an auxiliary loss L_{aux} to learn contextual representations of natural language. To be specific, we use the MLM loss (Devlin et al., 2019) as L_{aux} for transformer-based DST modes and the summation of both forward and backward LM losses (Peters et al., 2018) for RNN-based DST models. We only use the original MultiWOZ2.1 dialogs to optimize L_{aux}, considering that synthesized data is not suitable for natural language modelling. However, both the original and synthesized data are used to optimize L_{seq} and L_{cls}.

3.3 The Review Module

The process of review often help a learner consolidate difficult concepts newly learned. We design a *review* module to consider mispredicted examples as the concepts that the DST model has not grasped during CL, and utilize a schema-based data augmenter to produce similar cases from the examples. Specifically, the DST model is monitored at each stage of the CL training process. If a model is not converged at the end of an epoch in a training stage, we choose the top 10% incorrectly predicted examples according to their training losses as the resource to enlarge the cumulative dataset. The schema-based data augmenter uses three practical techniques to generate data as follows:

Slot Substitution. A mentioned slot name in (R_t, U_t) is changed into another slot name when its value is *dontcare*. Specifically, we first collect a word set for each slot name, e.g. {*'arrive', 'arriving', 'arrived'*} for the slot *'taxi-arriveby'*. Then, for a dialog data d_t where Y_t contains a slot s with the value *dontcare*, we substitute the word of s in the utterance with some word of another slot s' that is of the same domain and not mentioned in Y_t.

Value Replacement. A slot's value is replaced with another proper one when the value is explicitly contained in U_t. Specifically, we leverage the predefined schema in the dialog dataset to produce a value set for each slot and use the label map in (Heck et al., 2020) to figure out the position of value span within the utterance. The target value is then replaced with another one of the same slot.

Dialog Recombination. To recombine the dialog data d_t, we randomly search another dialog data in D that possesses the same mentioned slots (whose values are not *none*) in Y_t. We then cut and stitch their history X_{t-1} and current utterances (R_t, U_t), and exchange their Y_t to produce two new dialog data.

4 Experiments

Two popular datasets, WOZ2.0 (Wen et al., 2017) and MultiWOZ2.1 (Eric et al., 2020), are used to verify our approach. WOZ2.0 is a single-domain dataset with 1,200 dialogs and 3 slots. Multi-WOZ2.1 is a multi-domain dialog dataset with 10,438 dialogs, where there are 30 slots spanning 7 domains. The data splits (train/valid/test) of WOZ2.0 and MultiWOZ2.1 are 600/200/400 and 8438/1000/1000, respectively. We use the joint goal accuracy (JGA), the ratio of dialog data whose Z_t is correct, as the evaluation metric. We apply SaCLog onto TripPy (Heck et al., 2020), a transformer-based DST model, and TRADE (Wu et al., 2019), an RNN-based DST model, to show its effect. The slot encoder and the dialog context encoder are weight-shared. We use a BERT_{base} as the encoder and the `[CLS]` embedding as the slot embedding in TripPy, and use a bi-GRU as the encoder and the concatenation of the first and last hidden state as the slot embedding for TRADE. We also follow TripPy to add 2 new slot operations (i.e. *refer/dontcare*) into the classification types of L_{cls}.

Implementation Details. For the preview module, we use Adam (Kingma and Ba, 2015) with a fixed learning rate 3e-5 for 3 epochs in the pre-training. The batch size for L_{aux} is 14 and the batch size for L_{seq} and L_{cls} is 64. For the curriculum module, we perform a warm-up strategy for Adam optimizer with a maximum learning rate 1e-4. Before CL, we train models on full dataset for 2 epochs. After all subsets are accumulated, we then train for 10 extra epochs with a minimum learning rate 1e-6. We set the bucket number $N = 10$ and the crossed fold $K = 5$. The batch size is 36 and

Models	MultiWOZ2.1	WOZ2.0
GLAD (Zhong et al., 2018)	35.57%**	88.1±0.4%
SUMBT (Lee et al., 2019)	46.65%**	91.0±1.0%
DST-picklist (Zhang et al., 2019)	53.30%	–
Trippy (Heck et al., 2020)	55.29±0.28%	92.7±0.2%
SimpleTOD(Hosseini-Asl et al., 2020)	55.72%	–
CHAN (Shan et al., 2020)	58.55%	–
TripPy + ConvBERT	58.70%	93.1±0.3%*
TripPy + CoCoAug	60.53%	–
TripPy + SaCLog	**60.61±0.31%**	**94.2±0.2%**

Table 1: DST Results on MultiWOZ2.1 and WOZ2.0 in JGA. ∗ Our implemetation. ∗∗ MultiWOZ2.0 results.

Models	JGA
TripPy (ours)	58.17±0.25%
+ CL (rule-based)	58.38±0.17%
+ CL (model-based)	58.71±0.21%
+ CL (hybrid)	**58.85±0.23%**
+ SaCLog (w/o. review)	60.19±0.26%
+ SaCLog (w/o. preview)	60.23±0.34%
+ SaCLog	**60.61±0.31%**

Table 2: Ablation results on MultiWOZ2.1. +CL means adding the curriculum module only.

the maximum length is 256. To simplify the review process, we conduct data augmentation after the CL training is finished.

4.1 Performance of TripPy+SaCLog

Tables 1 shows the results of our approach comparing to various baselines. Based upon TripPy, we obtain state-of-the-art performance on both datasets with SaCLog. The two closest baselines[2], ConvBERT (Mehri et al., 2020) and CoCoAug (Li et al., 2021), are also built upon TripPy, where ConvBERT enhances its performance by using external large-scale conversational corpora to pre-train a BERT_{base} and CoCoAug leverages a delicate counter-factual augmentation skill to produce much larger training data. Our method, however, benefits from the CL framework and improves TripPy by utilizing the preview and review modules.

Ablation Study To examine how SaCLog facilitates DST training, we conduct detailed ablation experiments on MultiWOZ2.1, as shown in Table 2. In our re-implementation, we improve the basic TripPy by around 3% JGA via training for longer epochs (30 vs.10) and pre-training a BERT_{base} on MultiWOZ2.1 corpus only with the MLM loss. First, we investigate the influence of

---

[2]We implemented SaCLog upon these two methods, but no significant gains are observed. We conjecture that this is due to SaCLog has already largely exploited TripPy's potential so that the additional improvement of the two methods is limited.

Model	MultiWOZ2.1	WOZ2.0
TRADE	45.6%*	88.3±0.6%
+ SaCLog	**49.3±0.5%**	**91.1±0.4%**

Table 3: Results of TRADE+SaCLog on MultiWOZ2.1 and WOZ2.0. * Reported in (Eric et al., 2020).

difficulty scores by adding the curriculum module and utilizing the same pre-trained $BERT_{base}$. As we can see, using the hybrid difficulty score achieves better JGA (58.85%) than using either single score, indicating that both model prediction and human knowledge are necessary. When incorporating the other two modules in the CL framework, the performance is greatly boosted further. The combination of both modules increases the JGA by 1.76%, suggesting that the schema-aware pretraining and dialog augmentation are crucial for improving DST performance in the CL training.

4.2 Performance of TRADE+SaCLog

We also apply SaCLog to the classical RNN-based generative DST model, TRADE. As Table 3 shows, SaCLog improves TRADE by around 3∼4% JGA on both datasets, demonstrating the effectiveness of SaCLog on different types of base DST models.

5 Related Work

Curriculum Learning (CL) has attracted increasing research interests in various NLP tasks, such as machine translation (Liu et al., 2020; Zhou et al., 2020), general language understanding (Xu et al., 2020), reading comprehension (Tay et al., 2019) and open-domain chatbots (Bao et al., 2020; Cai et al., 2020; Su et al., 2020). Yet, the research on using CL in task-oriented dialog systems is limited. There has been some work (Saito, 2018; Zhao et al., 2021) on using CL in dialog policy learning, but applying CL to DST has not been investigated.

Learning a structural inductive bias during pretraining has been shown beneficial in downstream tasks that require parsing semantics, such as text-to-SQL (Yu et al., 2021) and table cell recognition (Wang et al., 2020). There are also many works (Hou et al., 2018; Yoo et al., 2020; Yin et al., 2020) on dialog augmentation. We aim to integrate these methods to build a general CL framework for DST.

6 Conclusion

In this paper, we propose a model-agnostic framework named as schema-aware curriculum learning for DST, which exploits both the curriculum

structure and the schema structure in task-oriented dialogs and shows to substantially improve DST performances. In the future, we plan to investigate CL approaches on other dialog modeling tasks.

Acknowledgments

References

Siqi Bao, Huang He, Fan Wang, Hua Wu, Haifeng Wang, Wenquan Wu, Zhen Guo, Zhibin Liu, and Xinchao Xu. 2020. Plato-2: Towards building an open-domain chatbot via curriculum learning. *arXiv preprint arXiv:2006.16779*.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48.

Hengyi Cai, Hongshen Chen, Cheng Zhang, Yonghao Song, Xiaofang Zhao, Yangxi Li, Dongsheng Duan, and Dawei Yin. 2020. Learning from easy to complex: Adaptive multi-curricula learning for neural dialogue generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7472–7479.

Giovanni Campagna, Agata Foryciarz, Mehrad Moradshahi, and Monica Lam. 2020. Zero-shot transfer learning with synthesized data for multi-domain dialogue state tracking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 122–132, Online. Association for Computational Linguistics.

Lu Chen, Boer Lv, Chi Wang, Su Zhu, Bowen Tan, and Kai Yu. 2020. Schema-guided multi-domain dialogue state tracking with graph attention neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7521–7528.

Yinpei Dai, Zhijian Ou, Dawei Ren, and Pengfei Yu. 2018. Tracking of enriched dialog states for flexible conversational information access. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6139–6143. IEEE.

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.

Shuyang Gao, Abhishek Sethi, Sanchit Agarwal, Tagyoung 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.

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.

Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. A simple language model for task-oriented dialogue. *Thirty-forth Conference on Neural Information Processing Systems*.

Yutai Hou, Yijia Liu, Wanxiang Che, and Ting Liu. 2018. Sequence-to-sequence data augmentation for dialogue language understanding. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1234–1245, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Jiaying Hu, Yan Yang, Chencai Chen, Liang He, and Zhou Yu. 2020. SAS: Dialogue state tracking via slot attention and slot information sharing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6366–6375, Online. Association for Computational Linguistics.

Sungdong Kim, Sohee Yang, Gyuwan Kim, and Sang-Woo Lee. 2020. Efficient dialogue state tracking by selectively overwriting memory. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 567–582, Online. Association for Computational Linguistics.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *International Conference on Learning Representations*.

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

Shiyang Li, Semih Yavuz, Kazuma Hashimoto, Jia Li, Tong Niu, Nazneen Rajani, Xifeng Yan, Yingbo Zhou, and Caiming Xiong. 2021. Coco: Controllable counterfactuals for evaluating dialogue state trackers. *International Conference on Learning Representations*.

Xuebo Liu, Houtim Lai, Derek F. Wong, and Lidia S. Chao. 2020. Norm-based curriculum learning for neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 427–436, Online. Association for Computational Linguistics.

Shikib Mehri, Mihail Eric, and Dilek Hakkani-Tur. 2020. Dialoglue: A natural language understanding benchmark for task-oriented dialogue. *arXiv preprint arXiv:2009.13570*.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Atsushi Saito. 2018. Curriculum learning based on reward sparseness for deep reinforcement learning of task completion dialogue management. In *Proceedings of the 2018 EMNLP Workshop SCAI: The 2nd International Workshop on Search-Oriented Conversational AI*, pages 46–51, Brussels, Belgium. Association for Computational Linguistics.

Yong Shan, Zekang Li, Jinchao Zhang, Fandong Meng, Yang Feng, Cheng Niu, and Jie Zhou. 2020. A contextual hierarchical attention network with adaptive objective for dialogue state tracking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6322–6333, Online. Association for Computational Linguistics.

Valentin I. Spitkovsky, Hiyan Alshawi, and Daniel Jurafsky. 2010. From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 751–759, Los Angeles, California. Association for Computational Linguistics.

Yixuan Su, Deng Cai, Qingyu Zhou, Zibo Lin, Simon Baker, Yunbo Cao, Shuming Shi, Nigel Collier, and Yan Wang. 2020. Dialogue response selection with hierarchical curriculum learning. *59th Annual Meeting of the Association for Computational Linguistics*.

Yi Tay, Shuohang Wang, Anh Tuan Luu, Jie Fu, Minh C. Phan, Xingdi Yuan, Jinfeng Rao, Siu Cheung Hui, and Aston Zhang. 2019. Simple and effective curriculum pointer-generator networks for read-

ing comprehension over long narratives. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4922–4931, Florence, Italy. Association for Computational Linguistics.

Xin Wang, Yudong Chen, and Wenwu Zhu. 2021. A survey on curriculum learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.

Zhiruo Wang, Haoyu Dong, Ran Jia, Jia Li, Zhiyi Fu, Shi Han, and Dongmei Zhang. 2020. Structure-aware pre-training for table understanding with tree-based transformers. *arXiv preprint arXiv:2010.12537*.

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

Jason Williams, Antoine Raux, and Matthew Henderson. 2016. The dialog state tracking challenge series: A review. *Dialogue & Discourse*.

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.

Peng Wu, Bowei Zou, Ridong Jiang, and AiTi Aw. 2020. GCDST: A graph-based and copy-augmented multi-domain dialogue state tracking. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1063–1073, Online. Association for Computational Linguistics.

Benfeng Xu, Licheng Zhang, Zhendong Mao, Quan Wang, Hongtao Xie, and Yongdong Zhang. 2020. Curriculum learning for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6095–6104, Online. Association for Computational Linguistics.

Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, and Qun Liu. 2020. Dialog state tracking with reinforced data augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9474–9481.

Kang Min Yoo, Hanbit Lee, Franck Dernoncourt, Trung Bui, Walter Chang, and Sang-goo Lee. 2020. Variational hierarchical dialog autoencoder for dialog state tracking data augmentation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3406–3425, Online. Association for Computational Linguistics.

Tao Yu, Chien-Sheng Wu, Xi Victoria Lin, Bailin Wang, Yi Chern Tan, Xinyi Yang, Dragomir Radev, Richard Socher, and Caiming Xiong. 2021. Grappa: Grammar-augmented pre-training for table semantic parsing. *International Conference on Learning Representations*.

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*.

Yangyang Zhao, Zhenyu Wang, and Zhenhua Huang. 2021. Automatic curriculum learning with over-repetition penalty for dialogue policy learning. *AAAI*.

Victor Zhong, Caiming Xiong, and Richard Socher. 2018. Global-locally self-attentive encoder for dialogue state tracking. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1458–1467, Melbourne, Australia. Association for Computational Linguistics.

Yikai Zhou, Baosong Yang, Derek F. Wong, Yu Wan, and Lidia S. Chao. 2020. Uncertainty-aware curriculum learning for neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6934–6944, Online. Association for Computational Linguistics.

Su Zhu, Jieyu Li, Lu Chen, and Kai Yu. 2020. Efficient context and schema fusion networks for multi-domain dialogue state tracking. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 766–781, Online. Association for Computational Linguistics.