Class Incremental Learning for Task-Oriented Dialogue System with Contrastive Distillation on Internal Representations (Student Abstract)

Qiancheng Xu[1], Min Yang[2], Binzong Geng[3]
[1]Georgia Institute of Technology
[2]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
[3]University of Science and Technology of China
qxu309@gatech.edu, min.yang@siat.ac.cn, bzgeng@mail.ustc.edu.cn

Abstract

The ability to continually learn over time by grasping new knowledge and remembering previously learned experiences is essential for developing an online task-oriented dialogue system (TDS). In this paper, we work on the class incremental learning scenario where the TDS is evaluated without specifying the dialogue domain. We employ contrastive distillation on the intermediate representations of dialogues to learn transferable representations that suffer less from catastrophic forgetting. Besides, we provide a dynamic update mechanism to explicitly preserve the learned experiences by only updating the parameters related to the new task while keeping other parameters fixed. Extensive experiments demonstrate that our method significantly outperforms the strong baselines.

Introduction

Task-oriented dialog systems (TDSs) allow users to seek information and complete complex tasks using natural language in an interactive manner. Remarkable success has been achieved in building end-to-end TDSs with sequence-to-sequence (Seq2Seq) models (Wu, Socher, and Xiong 2019). However, those models are achieved in a static environment which would suffer from significant performance degradation on past tasks after learning a new one, when encountering continuous streams of domains or functionalities through time. Therefore, it is necessary to equip a TDS with continual learning (CL) ability, through which the TDSs could continually learn over time by grasping new knowledge and retaining previously learned experiences. Recently, Geng et al. (2021) proposed a CL method for TDSs in the task incremental learning (TIL) scenario, assuming all task IDs accessible at inference time. However, the scenario simplifies the problem since it is possible to train models with task-specific components (van de Ven and Tolias 2019). In this paper, we focus on a more realistic scenario called class incremental learning (CIL) where task IDs are not given at test time. We propose a contrastive distillation on internal representations (TCDR) method for TDSs.

Implementation

Taking the dialogue history X and KB tuples B as inputs, TDS is expected to generate the system response Y word by word. Suppose a class-incremental TDS model has been built after learning a sequence of tasks from 1 to $t-1$, denoted as $M_{1:t-1}$. The goal of TDS in class-incremental learning scenario is to train a model $M_{1:t}$ that can handle all t tasks without knowing the task ID of the test data. For each encountered new task, we store R training samples into the memory buffer $\hat{D}$. $M_{1:t}$ will be trained on both the current data D and $\hat{D}$. We deploy GLMP (Wu, Socher, and Xiong 2019) as the base model given its effectiveness and efficiency for TDSs, which contains a global memory encoder, a local memory decoder and external knowledge (Fig 1).

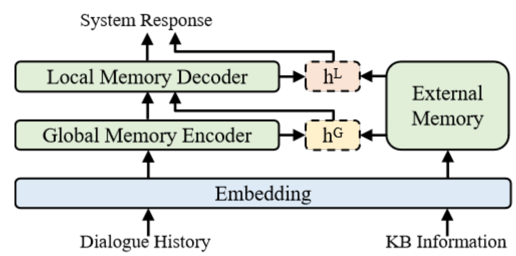

Figure 1: The base model of our TCDR method.

We first define a cross-entropy loss $\mathcal{L}_{\mathrm{CE}}$ for $M_{1:t}$ and a knowledge distillation loss $\mathcal{L}_{\mathrm{KD}}$ as follows:

$$\mathcal{L}_{\mathrm{CE}} = -\frac{1}{N}\sum_{i=1}^{N} y_i \log g(z_i) \tag{1}$$

$$\mathcal{L}_{\mathrm{KD}} = -\frac{1}{N}\sum_{i=1}^{N} g(\hat{z}_i/\tau) \log g(z_i/\tau) \tag{2}$$

where the ground truth label is denoted as y, while z and $\hat{z}$ are output logits of $M_{1:t}$ and $M_{1:t-1}$ respectively. N is the batch size, τ is the temperature scalar and $g(\cdot)$ is the softmax function. $\mathcal{L}_{\mathrm{KD}}$ encourages $M_{1:t}$ to produce similar distributional outputs with $M_{1:t-1}$:

Contrastive Distillation on Internal Representations We select h^G (initialized with the concatenation of the encoded dialogue history and KB information) and h^L (the last hop vector in the local memory decoder after querying the external knowledge) as internal representations, which could reflect all three components of GLMP comprehensively.

Task	Schedule	Navigation	Weather	Restaurant	Hotel	Attraction	CamRest	Avg.
GLMP	0.00/16.21	1.64/20.48	0.00/35.85	5.44/22.35	2.66/17.45	5.68/19.27	17.13/53.83	4.65/26.49
LwF	0.00/49.55	2.09/25.49	0.00/39.17	4.89/23.48	3.62/20.37	6.73/23.11	17.25/**56.48**	4.94/33.95
UCL	6.56/52.27	2.81/20.75	9.09/47.08	6.26/19.12	3.75/20.98	5.62/18.72	16.49/48.68	7.23/32.52
TPEM-CIL	0.00/46.37	1.04/**29.02**	0.00/53.00	2.46/18.61	1.22/15.13	2.67/18.65	10.84/51.69	2.60/33.21
Co2L	7.32/54.46	2.70/19.71	10.65/50.53	4.06/24.70	3.45/20.87	4.94/23.70	13.69/52.96	6.69/35.28
TCDR	**9.86/61.87**	**4.83**/20.86	**10.92/53.62**	**6.85/26.46**	**6.17/30.10**	**6.86/28.07**	**17.27**/54.20	**8.97/39.31**

Table 1: BLEU/Entity F1 results evaluated after all 7 tasks are visited. Avg. represents the average performance of all tasks.

We leverage the instances from $\hat{D}$ as anchors, while instances from D are merely used as negative samples. We pull together representations of the same sample produced by the old and new models and push apart the representations between different samples (Cha, Lee, and Shin 2021). Given each sample in $\hat{D}$, we treat its representations $\hat{h}^G$ and h^G which are learned by M_{t-1} and M_t as a positive pair, while the representations of samples in D are regarded as negative samples. Then the contrastive learning loss $\mathcal{L}_{\text{CL}}$ is:

$$\mathcal{L}_{\text{CL}} = -\frac{1}{N_1}\sum_{i=1}^{N_1}\log\frac{\exp(\hat{h}_i^G \cdot h_i^G/\rho)}{\exp(\hat{h}_i^G \cdot h_i^G/\rho)+\sum_{j=1}^{N_2}\exp(\hat{h}_i^G \cdot h_j^G/\rho)} \quad (3)$$

where ρ is the temperature hyperparameter. N_1 and N_2 are numbers of samples from $\hat{D}$ and D respectively. $\mathcal{L}_{\text{sim}}$ is calculated by comparing h^L and $\hat{h}^L$ of the same sample:

$$\mathcal{L}_{\text{sim}} = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{T} 1 - cos(h_{ij}^L, \hat{h}_{ij}^L) \quad (4)$$

where T is the target response length. N is the batch size where the training samples are selected from both D and $\hat{D}$. $cos()$ means cosine similarity. Using α, β and γ as hyperparameters, the overall loss $\mathcal{L}_{\text{overall}}$ can be defined as:

$$\mathcal{L}_{\text{overall}} = \mathcal{L}_{\text{CE}} + \alpha * \mathcal{L}_{\text{KD}} + \beta * \mathcal{L}_{\text{CL}} + \gamma * \mathcal{L}_{\text{sim}} \quad (5)$$

Dynamic Update for Previously Learned Model To alleviate representation inconsistency between tasks, we utilize a momentum strategy (He et al. 2020) to update $M_{1:t-1}$ according to the in-coming task. Instead of updating the whole model, we dynamically update parts of $M_{1:t-1}$ that are related to task t. Specifically, we use the embedding $C_{1:t}$ of $M_{1:t}$ to update parts of $C_{1:t-1}$ of $M_{1:t-1}$ while keeping other parts related to previous tasks be fixed. Mathematically, when learning task t, $C_{1:t-1}$ can be updated as:

$$C_{1:t-1}[j] = \begin{cases} C_{1:t-1}[j], & 1 \leq j \leq r \\ mC_{1:t}[j] + (1-m)C_{1:t-1}[j], & r < j \leq s \end{cases} \quad (6)$$

where m is a momentum coefficient and j denotes the embedding index of each input token. $\{1, ..., r\}$ are indexes of tokens for all previous tasks, while $\{r+1, ..., s\}$ are indexes of tokens for the current task. In this manner, $M_{1:t-1}$ could be adapted to new tasks to improve representation consistency, while keeping previously learned knowledge intact.

Experiments

Datasets Following (Geng et al. 2021), we evaluate TCDR on 7 tasks from 3 benchmark TDS datasets: Stanford multi-domain dialogue (SMD), Multi-WOZ 2.1, and CamRest, with 4,670/560/580 dialogues for training/validation/testing.

Hyperparameters The size of embedding and hidden state is 128. The model is trained using Adam optimizer with batch size of 32. We set τ, ρ, m to 2, 1, 1 respectively. The hyperparameters α, β, γ are all set to 1.

Baselines We compare TCDR with several methods: GLMP (trained on tasks separately without leveraging continual learning), LwF (Li and Hoiem 2018), UCL (Ahn et al. 2019), Co2L (Cha, Lee, and Shin 2021), and TPEM-CIL (Geng et al. 2021) in class incremental learning scenario.

Results We evaluate the performance of TDS with BLEU and entity F1 (Wu, Socher, and Xiong 2019). The results of each task after all 7 tasks have been learned are reported in Table 1. GLMP performs much worse than the continual learning methods (LwF, UCL and Co2L). This is consistent with our claim that conventional TDSs suffer from catastrophic forgetting. Meanwhile, TPEM-CIL achieves a poor result especially in BLEU scores, indicating that the CIL scenario appears to be much more challenging. In contrast, TCDR achieves better results than all the other baselines. This shows that our method could indeed preserve learned knowledge and adapt to new knowledge simultaneously.

References

Ahn, H.; Cha, S.; Lee, D.; and Moon, T. 2019. Uncertainty-based continual learning with adaptive regularization. In *NeurIPS*, 4392–4402.

Cha, H.; Lee, J.; and Shin, J. 2021. Co2L: Contrastive Continual Learning. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 9496–9505.

Geng, B.; Yuan, F.; Xu, Q.; Shen, Y.; Xu, R.; and Yang, M. 2021. Continual Learning for Task-oriented Dialogue System with Iterative Network Pruning, Expanding and Masking. In *ACL*.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum Contrast for Unsupervised Visual Representation Learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9726–9735.

Li, Z.; and Hoiem, D. 2018. Learning without Forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12): 2935–2947.

van de Ven, G. M.; and Tolias, A. S. 2019. Three scenarios for continual learning. *CoRR*, abs/1904.07734.

Wu, C.-S.; Socher, R.; and Xiong, C. 2019. Global-to-local Memory Pointer Networks for Task-Oriented Dialogue. *ArXiv*, abs/1901.04713.