Learning to Bridge Metric Spaces: Few-shot Joint Learning of Intent Detection and Slot Filling

Yutai Hou*, **Yongkui Lai***, **Cheng Chen**, **Wanxiang Che**†, **Ting Liu**
Research Center for Social Computing and Information Retrieval,
Harbin Institute of Technology
{ythou, yklai, cchen, car, tliu}@ir.hit.edu.cn

Abstract

In this paper, we investigate few-shot joint learning for dialogue language understanding. Most existing few-shot models learn a single task each time with only a few examples. However, dialogue language understanding contains two closely related tasks, i.e., intent detection and slot filling, and often benefits from jointly learning the two tasks. This calls for new few-shot learning techniques that are able to capture task relations from only a few examples and jointly learn multiple tasks. To achieve this, we propose a similarity-based few-shot learning scheme, named **Con**trastive **Pro**totype **M**erging network (**ConProm**), that learns to bridge metric spaces of intent and slot on data-rich domains, and then adapt the bridged metric space to specific few-shot domain. Experiments on two public datasets, Snips and FewJoint, show that our model significantly outperforms the strong baselines in one and five shots settings.

1 Introduction

Few-Shot Learning (FSL) that committed to learning new problems with only a few examples (Miller et al., 2000; Vinyals et al., 2016) is promising to break the data-shackles of current deep learning. Commonly, existing FSL methods learn a single few-shot task each time. But, real-world applications, such as dialogue language understanding, usually contain multiple closely related tasks (e.g., intent detection and slot filling) and often benefit from jointly learning these tasks (Worsham and Kalita, 2020; Chen et al., 2019; Qin et al., 2019; Goo et al., 2018). In few-shot scenarios, such requirements of joint learning present new challenges for FSL techniques to capture task relations from only a few examples and jointly learn multiple tasks.

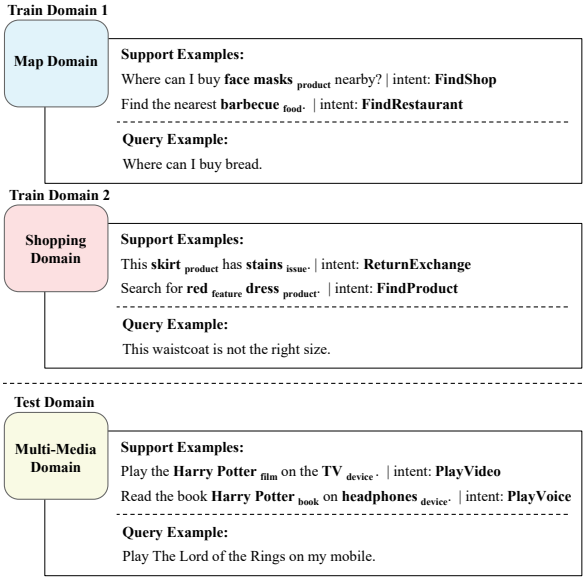

Figure 1: Examples of the few-shot joint dialogue language understanding. On each domain, given a few labeled support examples, the model predicts the intent and slot labels for unseen query examples. Joint learning benefits from capturing the relation between intent and slot labels, but such relation is hard to learn from a few sparse examples and hard to transfer across different domains.

This paper explores the few-shot joint learning in dialogue language understanding as an early attempt for this issue. As shown in Figure 1, FSL models are usually first trained on source training domains, then evaluated on an unseen target test domain. Although joint learning can improve dialogue language understanding by utilizing the relation between intents and slots, e.g., "Harry Potter" is "film" in "PlayVideo" intent and "book" in "PlayVoice" intent, it faces serious challenges when engaging to FSL setting. Firstly, it is hard to learn generalized intent-slot relations from only a few support examples. Secondly, because the intent-slot relation differs in different domains, it is

*Equal contributions.
†Corresponding author.

hard to directly transfer the prior experience from source domains to target domains. For instance, the intent-slot relation, "PlayVideo"-"film", has never appeared in source domains.

To tackle the aforementioned joint learning challenges in few-shot dialogue language understanding, we propose the **Prototype Merging**, which learns the intent-slot relation from data-rich training domains and adaptively captures and utilizes it to an unseen test domain. The intent-slot relation is learned with cross-attention between intent and slot class prototypes, which are the mean embeddings of the support examples belonging to the same classes. Such intent-slot relation adaptively connects the metric spaces of the two tasks.

Further, to jointly refine the intent and slot metric spaces bridged by Prototype Merging, we claim that related intents and slots, such as "PlayVideo" and "film", should be closely distributed in the metric space, otherwise, well-separated. To achieve this, we propose **Contrastive Alignment Learning**, which exploits class prototype pairs of related intents and slots as positive samples and non-related pairs as negative samples. With these samples, it regularizes the FSL process with a margined contrastive loss.

Overall, we named the above novel few-shot joint learning framework as **Con**trastive **Pro**totype **M**erging network (**ConProm**), which connects intent detection and slot filling tasks by bridging the metric spaces of them. Two main components of it cooperate to accomplish this goal. As shown in Figure 2, Prototype Merging builds the connection between two metric spaces, and Contrastive Alignment Learning refine the bridged metric space by properly distributing prototypes.

Experiments on two public datasets show both Prototype Merging and Contrastive Aligning Objective significantly boost the few-shot joint learning effects and outperform strong baselines. In summary, our contribution is three-fold: (1) We investigate the few-shot joint dialogue language understanding problem, which is also an early attempt for few-shot joint learning problem. (2) We propose a novel Prototype Merging mechanism to build intent-slot connections adaptively. (3) We introduce a Contrastive Alignment Learning objective to jointly refines the metric spaces of intent detection and slot filling. For reproducibility, our code for this paper is publicly available at https://github.com/AtmaHou/FewShotJoint.

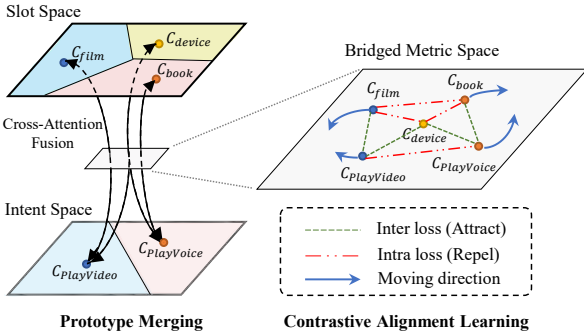

Figure 2: Illustration of two main components of the ConProm model: Prototype Merging and Contrastive Alignment Learning. C denotes prototypes. To ease understanding, we omit the repelling Inter loss in Bridged Metric Space, e.g, loss between C_{book} and $C_{\text{PlayVideo}}$.

2 Background

Before start, we introduce the background of dialogue language understanding and few-shot learning.

2.1 Dialogue Language Understanding

Dialogue language understanding contains two main components: intent detection and slot filling (Young et al., 2013). Intent detection is a sentence-level classification problem that classifies a user utterance into one of N intent categories.

Different from intent detection, slot filling aims to extract key entities within user utterances, which is often achieved by assigning slot tags to each token of a user utterance and is usually formulated as a sequence labeling problem. Given input utterance $\boldsymbol{x} = \langle x_1, x_2, \ldots, x_n \rangle$ as a sequence of words, joint dialogue language understanding predicts the corresponding semantic frame $\boldsymbol{y} = (l, \boldsymbol{t})$, where l is the intent label and $\boldsymbol{t} = \langle t_1, t_2, \ldots, t_n \rangle$ is the slot tags sequence of the utterance.

2.2 Few-shot Learning

Few-shot learning (FSL) extracts prior experience that allows quick adaption to new problems. Therefore, FSL models are usually first trained on a set of source domains, then evaluated on another set of unseen target domains. Figure 1 shows an example of the training and testing process of few-shot learning for dialogue language understanding.

A target domain only contains a few labeled examples, which is called support set $\mathcal{S} = \left\{(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)})\right\}_{i=1}^{|\mathcal{S}|}$. $\mathcal{S}$ includes K examples (K-shot) for each of N classes (N-way). Taking classification problem as an instance: given an input query

example $\boldsymbol{x} = \langle x_1, x_2, \ldots, x_n \rangle$ and a K-shot support set $\mathcal{S}$ as references, we find the most appropriate class y^* of $\boldsymbol{x}$:

$$y^* = \arg\max_y \; p(y \mid \boldsymbol{x}, \mathcal{S}).$$

State-of-the-art few-shot learning is often similarity-based methods (Bao et al., 2020; Snell et al., 2017). These methods conquer the extreme lack of data by learning a general similarity metric space on data-rich source domains. Then on few-shot target domains, they classify a query example according to example-class similarity, where class representations are obtained from a few support examples.

Prototypical network (Snell et al., 2017) is one of the most classical similarity-based methods. It obtains the class representation as to the mean embedding of support examples belonging to the same class, so called **prototypes**:

$$C_i = \frac{1}{|\mathcal{S}_i|} \sum_{(\boldsymbol{x},y)\in\mathcal{S}_i} E(\boldsymbol{x}),$$

where $\mathcal{S}_i$ is the set of support examples of the ith class, and $E(\cdot)$ is the embedding function. The probability of $\boldsymbol{x}$ belongs to the ith class is then made as:

$$p(y_i \mid \boldsymbol{x}, S) = \frac{\exp\left(\textsc{Sim}(E(\boldsymbol{x}), C_i)\right)}{\sum_j \exp\left(\textsc{Sim}(E(\boldsymbol{x}), C_j)\right)},$$

where $\textsc{Sim}(\cdot,\cdot)$ is a vector similarity function.

3 Proposed Method

In this section, we introduce the proposed **Con**trastive **Pro**totype **M**erging network (ConProm). Firstly, we describe the few-shot intent detection and slot filling with Prototypical network (§3.1). Based on that, we present two key components of ConProm: the **Prototype Merging** mechanism that adaptively connects two metric spaces of intent and slot (§3.2) and the **Contrastive Alignment Learning** that jointly refines the metric space connected by Prototype Merging (§3.3).

3.1 Few-shot Intent Detection and Slot Filling

We build our few-shot intent detection and slot filling model based on the Prototypical Network described in Section 2.2. Given a query sentence $\boldsymbol{x}$ and a support set $\mathcal{S}$, we estimate the probability of $\boldsymbol{x}$ being associated with intent label l_i as:

$$p(l_i \mid \boldsymbol{x}, \mathcal{S}) = \frac{\exp\left(\textsc{Sim}(E_{\text{intent}}(\boldsymbol{x}), C_{\text{intent}_i})\right)}{\sum_j \exp\left(\textsc{Sim}(E_{\text{intent}}(\boldsymbol{x}), C_{\text{intent}_j})\right)},$$

and estimates the probability of the kth token in $\boldsymbol{x}$ belonging to the ith slot class as:

$$p(t_i \mid k, \boldsymbol{x}, \mathcal{S}) = \frac{\exp\left(\textsc{Sim}(E_{\text{slot}}(x_k), C_{\text{slot}_i})\right)}{\sum_j \exp\left(\textsc{Sim}(E_{\text{slot}}(x_k), C_{\text{slot}_j})\right)},$$

where C_{intent_i} and C_{slot_i} are prototypes derived with support examples. $E_{\text{intent}}(\cdot)$ and $E_{\text{slot}}(\cdot)$ are embedder functions for intent and slot respectively. We adopt BERT (Devlin et al., 2019) as the embedder, and the sentence embedding $E_{\text{intent}}(\boldsymbol{x})$ is calculated as the averaged embedding of its tokens. We use the dot-product similarity for function $\textsc{Sim}(\cdot,\cdot)$.

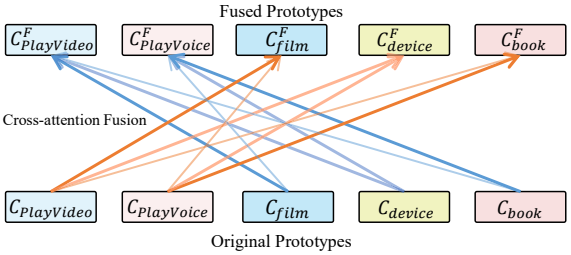

Figure 3: Illustration of cross-attention based information fusion in Prototype Merge. Thicker lines indicate higher cross-attention scores. For example, "PlayVideo" and "film" are more related, so the corresponding score is larger.

3.2 Prototype Merging

To achieve few-shot joint learning and capture the intent-slot relation with the similarity-based method described above, we need to bridge the metric spaces of intent detection and slot filling. However, as mentioned in the introduction, intent-slot relation differs in different domains, it is hard to transfer the bridged metric space learned from source domains to target domains.

To remedy this, we propose the **Prototype Merging** that can bridge metric spaces adaptively. As shown in Figure 3, Prototype Merging adaptively estimates intent-slot relevance with cross-attention between intent and slot, and then merges the intent and slot prototypes with attentive information fusion. Such an attentive fusion process enables both intent and slot prototype representations to reflect intent-slot relation and improves domain transferability.

On an unseen target domain, we estimate the intent-slot cross-attention scores from the support set with two methods: (1) use the statistic of co-occurrence of different intents and slots; (2) estimate the intent-slot relevance score using prototype representations.

Firstly, for the statistic-based attention-score, we estimate intent-slot attention scores A^{S} by counting the co-occurrence of different intents and slots, where $A^{\mathrm{S}}_{i,j}$ records the normalized number of co-occurrence times for the ith intent and the jth slot (normalized by row).

Secondly, for representation-based attention-score, we estimate the cross-attention scores with the Additive Attention (Bahdanau et al., 2015):[1]

$$A^{\mathrm{R}}_{i,j} = V^{\top}\tanh(WC_{\mathrm{intent}_i} + UC_{\mathrm{slot}_j}),$$

where A^{R} is the attention matrix, and $A^{\mathrm{R}}_{i,j}$ records the cross-attention score between the ith intent and the jth slot. U, V and W are parameters learned on source domains,which preserve the general experience of estimating relevance with representations. C_{intent_i} and C_{slot_j} are prototypes of ith intent and the jth slot respectively. We normalize A^{R} by row with softmax function.

We obtain the final cross-attention score matrix A by combining A^{S} and A^{R}.

$$A = \lambda A^{\mathrm{S}} + (1-\lambda)A^{\mathrm{R}},$$

where λ is the interpolation factor.

After obtaining the cross-attention scores, we represent each intent by fusing the information of related slot prototypes, where the attention scores are used as fusing weights. Similarly, we use intent prototypes to represent slots (See Figure 3). The fusion process is as follows:

$$C^{\mathrm{F}}_{\mathrm{intent}_i} = \sum_j A_{ij} \times C_{\mathrm{slot}_j},$$
$$C^{\mathrm{F}}_{\mathrm{slot}_j} = \sum_i A_{ij} \times C_{\mathrm{intent}_i},$$

where $C^{\mathrm{F}}_{\mathrm{intent}_i}$ and $C^{\mathrm{F}}_{\mathrm{slot}_j}$ are the fused prototypes of ith intent and the jth slot respectively.

At last, we obtain the representation of merged prototypes C' by combining the origin prototype C with the fused prototype C^{F}:

$$C'_{\mathrm{intent}} = \alpha \times C^{\mathrm{F}}_{\mathrm{intent}} + (1-\alpha) \times C_{\mathrm{intent}},$$
$$C'_{\mathrm{slot}} = \alpha \times C^{\mathrm{F}}_{\mathrm{slot}} + (1-\alpha) \times C_{\mathrm{slot}},$$

where the α is a hyper-parameter that controls the importance of intent-slot relation.

3.3 Contrastive Alignment Learning

Similarity-based few-shot learning relies heavily on a good metric space, where different classes should be well separated from each other (Hou et al., 2020a; Yoon et al., 2019). In joint-learning scenarios, there are further requests to connect metric spaces of joint learned tasks and jointly optimize these metric spaces.

In response to the above requests, we argue that the distribution of prototypes of dialogue language understanding should fit these intuitions: (1) different intent prototypes should be far away and the same as slot prototypes (*Intra-Contrastive*); (2) the slot prototypes should close to the related intent prototypes and should be far away from the unrelated intent prototypes (*Inter-Contrastive*).[2] To achieve these, we introduce a **Margined Contrastive Loss** to force the model to learn the separation and alignment of intent and slot prototypes.

Firstly, to encourage separation of prototypes from the same task, we regularize the learning of intent and slot prototypes with *Intra-Contrastive* loss $\mathcal{L}_{\mathrm{Intra}} = \frac{1}{2}(\mathcal{L}_{\mathrm{Intra-intent}} + \mathcal{L}_{\mathrm{Intra-slot}})$, where both the $\mathcal{L}_{\mathrm{Intra-intent}}$ and $\mathcal{L}_{\mathrm{Intra-slot}}$ are calculated as:

$$\mathcal{L}_{\mathrm{Intra}} = \frac{1}{N^2}\sum_i\sum_j \max(0, m - \|C_i - C_j\|)^2,$$

where m is the margin value and N is the number of prototypes. The margin m is important since it can protect metric space from excessive dispersion.

Next, we learn the alignment (separation) between intent prototypes and slot prototypes with *Inter-Contrastive* loss $\mathcal{L}_{\mathrm{Inter}}$:

$$\mathcal{L}^{\mathcal{R}}_i = \frac{1}{2|\mathcal{R}_i|}\sum_{j\in\mathcal{R}_i}(\left\|C_{\mathrm{intent}_i} - C_{\mathrm{slot}_j}\right\|^2),$$
$$\mathcal{L}^{\mathcal{U}}_i = \frac{1}{2|\mathcal{U}_i|}\sum_{k\in\mathcal{U}_i}\max(0, m - \|C_{\mathrm{intent}_i} - C_{\mathrm{slot}_k}\|)^2,$$
$$\mathcal{L}_{\mathrm{Inter}} = \sum_i^{N_I}(\mathcal{L}^{\mathrm{R}}_i + \mathcal{L}^{\mathrm{U}}_i),$$

[1] We adopt additive attention because we find it outperforms common product-based attention in our setting. This is mainly due to that additive attention interferes less with product-based similarity calculations.

[2] A slot is related to an intent means that they used to co-occur in the same semantic frame.

where $\mathcal{R}_i$ is the set of slots related to the ith intent and $\mathcal{U}_i$ is the set of slots that are not related to the ith intent. N_I is the number of intents. Here, we simply obtain the relatedness with the co-occurrence matrix M^{S} in Section 3.2.

Finally, the Margin Contrastive Loss is calculated as:

$$\mathcal{L}_{\text{Contrastive}} = \mathcal{L}_{\text{Inter}} + \mathcal{L}_{\text{Intra}}$$

3.4 Learning Objective

In dialogue language understanding task, we joint learn the intent detection task and slot filling by optimizing both losses at the same time. Specifically, we use CrossEntropy (CE) to calculate the loss for intent detection and slot filling. Combining with the loss of Contrastive Alignment Learning, we train the entire model with the following objective function:

$$\mathcal{L}_{all} = \mathrm{CE}_{\text{intent}} + \mathrm{CE}_{\text{slot}} + \mathcal{L}_{\text{Contrastive}}$$

4 Experiments

We evaluate our method on the dialogue language understanding task of 1-shot/5-shot setting, which transfers knowledge from source domains (training) to an unseen target domain (testing) containing only 1-shot/5-shot support set.

4.1 Settings

Dataset We conduct experiments on two public datasets: Snips (Coucke et al., 2018) and FewJoint (Hou et al., 2020c). Snips is a widely-used dataset for dialogue language understanding, containing seven single-intent domains together with 53 slots. The other dataset FewJoint is joint dialogue language understanding used in the few-shot learning contest of SMP2020-ECDT Task-1.[3] It contains 59 multi-intent domains, 143 different intents, and 205 different slots.

In the few-shot learning setting, we train models on several source domains and test them on unseen target few-shot domains. For Snips, we follow Krone et al. (2020a) and combine single-intent domain into multi-intent domain to achieve the classification of intents. After that, we split the Snips dataset into 3 parts: the training domain with 3 intents, the developing domain with 2 intents and the testing domain with 2 intents. FewJoint is already a few-shot learning benchmark. Therefore, we follow the original data split and there are 45 domains for training, 5 domains for developing and 9 domains for testing.

Few-shot Dataset Construction To simulate the few-shot learning situation, we follow previous few-shot learning works (Vinyals et al., 2016; Krone et al., 2020a; Finn et al., 2017) and construct the dataset into a few-shot episode style, where the model is trained and evaluated with a series of few-shot episodes. Each episode contains a support set and query set. However, different from the single-task problem, joint-learning examples are associated with multiple labels. Therefore, we cannot guarantee that each label appears K times while sampling examples for the K-shot support set. To remedy this, we build support sets with the Mini-Including Algorithm (Hou et al., 2020a), which is intended for such situations. It constructs support set generally following two criteria: (1) All labels appear at least K times in support set. (2) At least one label will appear less than K times in the support set if any support example is removed from the support set. For Snips, we construct 200 few-shot episodes for training, 50 for developing, and 50 for testing. We set the query set size as 16 for training and developing, 100 for testing. For FewJoint, we use the few-shot episodes provided by the original dataset.

Evaluation We adopt three metrics for evaluation: Intent Accuracy, Slot F1-score, Joint Accuracy.[4] For joint dialogue language understanding, Joint Accuracy is the most important metric among all three metrics (Hou et al., 2020c). It evaluates the sentence level accuracy, which considers one sentence is correct only when all its slots and intents are correct.

To conduct a robust evaluation under few-shot setting, we validate the models on multiple few-shot episodes (i.e., support-query set pairs) from different domains and take the average score as final results. To control the non-deterministic neural network training (Reimers and Gurevych, 2017), we report the average score of 5 random seeds for all results.

4.2 Baselines

We compare our model with two kinds of strong baseline: fine-tune based transfer learning methods

[3]The Eighth China National Conference on Social Media Processing https://smp2020.aconf.cn/smp.html

[4]We calculate the Slot F1-score with the conll-eval script https://www.clips.uantwerpen.be/conll2000/chunking/conlleval.txt

Models	Snips			FewJoint		
	Intent Acc.	Slot F1	Joint Acc.	Intent Acc.	Slot F1	Joint Acc.
SepProto	**98.23**±0.66	43.90±1.98	9.47±2.10	66.35±0.51	27.24±1.10	10.92±0.89
JointProto	92.57±0.57	42.63±2.03	7.35±1.70	58.52±0.28	29.49±1.01	9.64±0.47
LD-Proto	97.25±0.71	47.81±2.53	10.67±1.99	**67.70**±0.65	27.73±0.35	13.70±0.52
LD-Proto+TR	97.53±0.30	51.03±2.40	17.32±2.62	67.63±1.42	34.06±4.75	16.98±2.14
ConProm (Ours)	96.67±1.45	53.05±0.81	21.72±0.97	65.26±0.23	33.09±1.66	16.32±0.75
ConProm+TR (Ours)	96.17±0.76	**55.84**±0.85	**29.72**±1.30	65.73±0.55	**37.97**±0.70	**19.57**±1.19
JointTransfer	71.07±4.31	38.24±2.19	13.28±0.45	41.83±2.40	26.89±2.72	12.27±2.09
Meta-JOSFIN	71.38±0.76	31.47±0.29	8.88±0.18	57.92±0.66	29.26±0.45	15.00±0.66
LD-Proto+FT	83.85±6.21	45.76±5.24	17.70±2.67	**64.70**±0.50	32.15±1.28	21.32±1.80
ConProm+FT (Ours)	88.20±3.22	52.41±2.01	23.05±1.70	61.24±0.81	42.02±0.77	24.63±1.30
ConProm+FT+TR (Ours)	**90.45**±0.52	**56.04**±1.75	**27.80**±2.33	63.67±0.94	**42.44**±0.51	**27.72**±0.95

Table 1: Scores on 1-shot dialogue language understanding task on Snips and FewJoint datasets. **+FT** denotes finetune model. **+TR** denotes using the trick of transition rule, which blocks illegal slot prediction, such as "I" tag after "O" tag. Results above the mid-line are from non-finetune based methods, and results below the mid-line are from finetuning based methods.

Models	Snips			FewJoint		
	Intent Acc.	Slot F1	Joint Acc.	Intent Acc.	Slot F1	Joint Acc.
SepProto	**99.53**±0.11	53.28±1.85	14.40±3.00	75.64±1.51	36.08±0.65	15.93±1.85
JointProto	99.17±0.09	50.63±2.01	13.40±1.44	70.93±2.45	39.47±1.05	14.48±1.11
LD-Proto	99.40±0.08	48.96±1.85	20.93±3.00	**78.29**±1.51	39.88±0.65	22.91±1.85
LD-Proto+TR	99.20±0.30	54.87±3.79	29.40±2.90	75.75±0.95	**51.62**±2.82	27.59±2.31
ConProm (Ours)	98.50±0.42	61.03±1.77	32.20±2.06	78.05±1.04	39.40±1.75	24.18±1.29
ConProm+TR (Ours)	98.99±0.14	**65.13**±1.46	**40.20**±2.24	75.54±1.85	50.28±1.03	**28.69**±1.61
JointTransfer	88.87±5.04	49.62±1.87	25.50±3.09	57.50±6.09	29.00±4.35	18.81±4.45
Meta-JOSFIN	92.47±1.26	56.85±1.25	25.87±0.31	78.91±0.53	53.88±1.63	36.63±1.01
LD-Proto+FT	81.07±8.61	59.27±3.61	26.33±2.38	**80.50**±0.97	55.33±2.55	38.11±2.60
ConProm+FT (Ours)	96.23±1.19	66.66±2.46	39.87±2.60	78.33±1.14	62.34±0.26	40.25±1.19
ConProm+FT+TR (Ours)	**98.40**±0.20	**72.98**±0.41	**52.95**±0.85	78.43±1.86	**69.44**±0.39	**46.54**±0.72

Table 2: Scores on 5-shot dialogue language understanding task on Snips dataset and FewJoint dataset.

(JointTransfer, Meta-JOSFIN) and similarity-based FSL methods (SepProto, JointProto, LD-Proto).

JointTransfer is a domain transfer model based on the JointBERT (Chen et al., 2019). It consists of a shared BERT embedder with intent detection and slot filling layers on the top. We pretrain it on source domains and finetune it on target domain support sets.

Meta-JOSFIN (Bhathiya and Thayasivam, 2020) is a meta-learning model based on the MAML (Finn et al., 2017). The meta-learner model here is a BERT-based joint dialogue language understanding model similar to **JointTransfer**. It learns initial parameters that can fast adapt to the target domain after only a few updates.

SepProto is a prototypical-based dialogue language understanding model with BERT embedding, that learns intent detection and slot filling separately. During the experiment, it is pre-trained on source domains and then directly applies to target domains without fine-tuning.

JointProto (Krone et al., 2020a) is all the same as SepProto except that it jointly learns the intent and slot tasks by sharing the BERT encoder.

LD-Proto is also a prototypical model similar to **JointProto**. The only difference is that it is enhanced by the logits-dependency tricks (Goo et al., 2018), where joint learning is achieved by depending on the intent and slot prediction on the logits of the accompanying task.

Implements For both ours and baseline models, we determine the hyperparameters on the development set. We use ADAM (Kingma and Ba, 2015) for training and set batch size as 4 and learning rate

as 10^{-5}. We adopt embedding tricks of Pairs-Wise Embedding (Gao et al., 2019; Hou et al., 2020a) and Gradual Unfreezing (Howard and Ruder, 2018). The λ and α in Section 3.2 are both set as 0.5. We implement both our and baseline models with the few-shot platform MetaDialog.[5] Besides, to use the information in target domains and make a fair comparison with fine-tuning baselines, we explore the performance of the similarity-based model under fine-tuning setting (+FT) and enhance the model with a fine-tune process similar to Meta-JOSFIN. In addition, following the suggestions of Hou et al. (2020a), we investigate adding Transition Rules (+TR) between slot tags, which bans illegal slot prediction, such as "I" tag after "O" tag.

4.3 Main Results

In this section, we present the evaluation of the proposed method on both 1-shot and 5-shot dialogue understanding setting.

Result of 1-shot setting As shown in Table 1, our method (ConProm) achieves the best performance on Joint Accuracy, which is the most important metric. Among all metrics, ConProm only lags a bit than LD-Proto on intent accuracy. We address this to the fact that there are many slots shared by different intent, and representing an intent with slots may unavoidably introduce noise from other intents. Considering the huge improvements on Slot and Joint performance over LD-Proto, we argue that the limited loss is a worthy compromise here. Since similarity-based models predict slot tags independently for each token, they tend to predict illegal tags. We employ a simple transition rule (+TR) to remedy such defects and further improves the performance. For fairness, we also enhance LD-Proto with TR trick and our model still outperforms the enhanced baseline.

For those non-finetuned methods, ConProm outperforms LD-Proto by Joint Accuracy scores of 11.05 on Snips and 2.62 on FewJoint, which show that our model can better capture the relation between intent and slot. Our improvements on Snips are higher than those on FewJoint, which is mainly because that there is clearer intent-slot dependency in Snips. The performance of JointProto is even lower than SepProto, which demonstrates that few-shot joint learning is not a trivial issue as simply sharing the embeddings

[5] https://github.com/AtmaHou/MetaDialog

Setting	Snips		FewJoint	
	1-shot	5-shot	1-shot	5-shot
Ours	21.72	32.20	16.32	24.18
- PM	-1.90	-2.63	-4.90	-8.39
- CAL	-5.19	-12.73	-1.78	-3.78

Table 3: Ablation study over two main components of proposed framework: Prototype Merge (PM) and Contrastive Alignment Learning (CAL). The score is Joint Accuracy.

When finetuning brings significant improvements for all methods, our model (ConProm+FT) still achieves the best performance. Interestingly, we observe that finetuning often hurts the intent prediction. This shows that finetuning brings limited gains on sentence-level domain knowledge but leads to overfitting.

Result of 5-shot setting Table 2 shows the 5-shot results. The results are consistent with 1-shot setting in general trending and our methods achieve the best performance. While more learning shots improve the performance for all methods, the superiority of our best performed baseline is further strengthened. This shows that the model can better exploit the richer intent-slot relations hidden in 5-shot support sets.

4.4 Analysis

Ablation Test To inspect how each component of the proposed model contributes to the final performance, we conduct ablation analysis. As shown in Table 3, we independently removing two main components: Prototype Merge (PM) and Contrastive Alignment Learning (CAL).

When PM is removed, the intent and slot prototypes are represented only with corresponding support examples, and Joint Accuracy drops are witnessed. There is more loss on FewJoint. Because there are much more slots shared by different intents in FewJoint, and the attention mechanism of PM is important for identifying relatedness between intents and slots.

For our model without CAL, we train the model with only cross entropy loss and get lower scores on all settings. There are more performance drops on Snips. This is mainly because that there much clearer intent-slot relation in Snips, which can be easily handled by CAL.

In terms of contribution, there are opposite performance for CAL and PM on two dataset, which shows that PM and CAL complement each other and reach a balance for various situations.

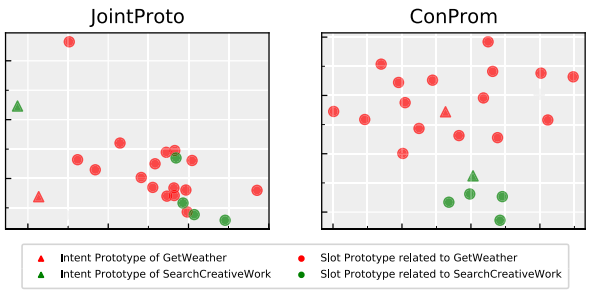

Figure 4: Visualization of the prototype distribution of JointProto and Ours (ConProm) with tSNE (step=500).

Models	Snips		FewJoint	
	F1.	Acc.	F1.	Acc.
JointProto	42.63	8.08	29.49	15.73
LD-Proto	47.81	10.72	27.73	20.44
LD-Proto+TD	51.03	17.53	34.06	24.69
ConProm	53.05	22.30	33.09	22.38
ConProm+TD	55.84	30.47	37.97	26.31
JointTransfer	38.24	14.38	26.89	26.37
Meta-JOSFIN	31.47	9.73	29.26	21.73
LD-Proto+FT	45.76	21.92	32.15	35.75
ConProm+FT	52.41	25.97	42.02	39.62
ConProm+TD+FT	56.04	27.91	42.44	40.71

Table 4: Analysis for sentence level slot accuracy.

Visual Analysis of Prototype Distribution To get further an understanding of the model effects on bridging the metric spaces of intent and slot, we visualize the prototype distributions in the metric space. As shown in Figure 4, it is exciting to see that our model successfully refine the prototype distribution by aligning the slots to related intent and making prototypes properly well-separated.

Sentence level slot accuracy analysis There is some confusion in Table 1 and Table 2 that there are huge performance differences of Joint Accuracy score when Intent Accuracy scores and Slot F1 scores are similar. We inspect this issue by evaluating the Sentence Level Slot Accuracy, which considers a sentence to be correct when all slots are correct. As shown in Table 4, there is a huge gap in the slot accuracy score between LD-Proto and ConProm, which explains the gap in Joint score.

5 Related Work

Few-shot learning is one of the most important direction for machine learning area (Fei-Fei, 2006; Fink, 2004) and often achieved by similarity-based method (Vinyals et al., 2016) and fine-tuning based method (Finn et al., 2017). FSL in natural language processing has been explored for various tasks, including text classification (Sun et al., 2019; Geng et al., 2019; Yan et al., 2018; Yu et al., 2018), entity relation classification (Lv et al., 2019; Gao et al., 2020; Ye and Ling, 2019), sequence labeling (Luo et al., 2018; Hou et al., 2018; Shah et al., 2019; Hou et al., 2020a; Liu et al., 2020).

As the important part of a dialog system, dialogue language understanding attract a lot of attention in few-shot scenario. Dopierre et al. (2020); Vlasov et al. (2018); Xia et al. (2018) explored few-shot intent detection technique. Luo et al. (2018) and Hou et al. (2020a) investigated few-shot slot tagging by using prototypical network. Hou et al. (2020b) explored few-shot multi-label intent detection with an adaptive logit adapting threshold. But all of these works focus on a single task.

Despite a lot of works on joint dialogue understanding (Goo et al., 2018; Li et al., 2018; Zhang et al., 2019; Qin et al., 2019; Wang et al., 2018; E et al., 2019; Wu et al., 2020; Gangadharaiah and Narayanaswamy, 2019; Liu et al., 2019; Qin et al., 2020), few-shot joint dialogue understanding is less investigated. Krone et al. (2020b) and Bhathiya and Thayasivam (2020) make the earliest attempts by directly adopt general and classic few-shot learning methods such as MAML and prototypical network. These methods achieve joint learning by sharing the embedding between intent detection and slot filling task, which model the relation between intent and slot task implicitly. By contrast, we explicitly model the interaction between intent and slot with attentive information fusion and constrastive loss. Experiment results also demonstrate the superiority of our method on this task.

6 Conclusion

In this paper, we propose a similarity-based few-shot joint learning framework, ConProm, for dialogue understanding. To adaptively model the interaction between intents and slots, we propose the Prototype Merging that bridges the intent metric and slot metric spaces with cross-attention between intent and slot. To learn better bridged metric space for intent and slot, we propose the Contrastive Alignment Learning to align related cross-task labels in metric space and force unrelated labels properly separated. Experiment results validate that both Prototype Merging and Contrastive Alignment Learning can improve performance.

Acknowledgments

We are grateful for the helpful comments and suggestions from the anonymous reviewers. This work was supported by the National Key R&D Program of China via grant 2020AAA0106501 and the National Natural Science Foundation of China (NSFC) via grant 61976072 and 61772153.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proc. of the ICLR*.

Yujia Bao, Menghua Wu, Shiyu Chang, and Regina Barzilay. 2020. Few-shot text classification with distributional signatures. In *Proc. of the ICLR*. OpenReview.net.

Hemanthage S Bhathiya and Uthayasanker Thayasivam. 2020. Meta learning for few-shot joint intent detection and slot-filling. In *Proc. of the ICMLT*, pages 86–92.

Qian Chen, Zhu Zhuo, and Wen Wang. 2019. BERT for joint intent classification and slot filling. *CoRR*, abs/1902.10909.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, Maël Primet, and Joseph Dureau. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *CoRR*, abs/1805.10190.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of the NAACL, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Thomas Dopierre, Christophe Gravier, Julien Subercaze, and Wilfried Logerais. 2020. Few-shot pseudo-labeling for intent detection. In *Proc. of the COLING*, pages 4993–5003, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Haihong E, Peiqing Niu, Zhongfu Chen, and Meina Song. 2019. A novel bi-directional interrelated model for joint intent detection and slot filling. In *Proc. of the ACL*, pages 5467–5471, Florence, Italy. Association for Computational Linguistics.

Li Fei-Fei. 2006. Knowledge transfer in learning to recognize visual objects classes. In *Proc. of the ICDL*, pages 1–8.

Michael Fink. 2004. Object classification from a single example utilizing class relevance metrics. In *Proc. of the NIPS*, pages 449–456.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proc. of the ICML*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR.

Rashmi Gangadharaiah and Balakrishnan Narayanaswamy. 2019. Joint multiple intent detection and slot labeling for goal-oriented dialog. In *Proc. of the NAACL*, pages 564–569, Minneapolis, Minnesota. Association for Computational Linguistics.

Tianyu Gao, Xu Han, Ruobing Xie, Zhiyuan Liu, Fen Lin, Leyu Lin, and Maosong Sun. 2020. Neural snowball for few-shot relation learning. In *Proc. of the AAAI*, pages 7772–7779. AAAI Press.

Tianyu Gao, Xu Han, Hao Zhu, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2019. FewRel 2.0: Towards more challenging few-shot relation classification. In *Proc. of the EMNLP-IJCNLP*, pages 6250–6255, Hong Kong, China. Association for Computational Linguistics.

Ruiying Geng, Binhua Li, Yongbin Li, Xiaodan Zhu, Ping Jian, and Jian Sun. 2019. Induction networks for few-shot text classification. In *Proc. of the EMNLP-IJCNLP*, pages 3904–3913, Hong Kong, China. Association for Computational Linguistics.

Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. 2018. Slot-gated modeling for joint slot filling and intent prediction. In *Proc. of the NAACL, Volume 2 (Short Papers)*, pages 753–757, New Orleans, Louisiana. Association for Computational Linguistics.

Yutai Hou, Wanxiang Che, Yongkui Lai, Zhihan Zhou, Yijia Liu, Han Liu, and Ting Liu. 2020a. Few-shot slot tagging with collapsed dependency transfer and label-enhanced task-adaptive projection network. In *Proc. of the ACL*, pages 1381–1393, Online. Association for Computational Linguistics.

Yutai Hou, Yongkui Lai, Yushan Wu, Wanxiang Che, and Ting Liu. 2020b. Few-shot learning for multi-label intent detection. *CoRR*, abs/2010.05256.

Yutai Hou, Yijia Liu, Wanxiang Che, and Ting Liu. 2018. Sequence-to-sequence data augmentation for dialogue language understanding. *Proc. of the COLING*.

Yutai Hou, Jiafeng Mao, Yongkui Lai, Cheng Chen, Wanxiang Che, Zhigang Chen, and Ting Liu. 2020c. Fewjoint: A few-shot learning benchmark for joint language understanding. *CoRR*.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proc. of the ACL (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proc. of the ICLR*.

Jason Krone, Yi Zhang, and Mona Diab. 2020a. Learning to classify intents and slot labels given a handful of examples. In *Proc. of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 96–108, Online. Association for Computational Linguistics.

Jason Krone, Yi Zhang, and Mona Diab. 2020b. Learning to classify intents and slot labels given a handful of examples. In *Proc. of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 96–108, Online. Association for Computational Linguistics.

Changliang Li, Liang Li, and Ji Qi. 2018. A self-attentive model with gate mechanism for spoken language understanding. In *Proc. of the EMNLP*, pages 3824–3833, Brussels, Belgium. Association for Computational Linguistics.

Yijin Liu, Fandong Meng, Jinchao Zhang, Jie Zhou, Yufeng Chen, and Jinan Xu. 2019. CM-net: A novel collaborative memory network for spoken language understanding. In *Proc. of the EMNLP-IJCNLP*, pages 1051–1060, Hong Kong, China. Association for Computational Linguistics.

Zihan Liu, Genta Indra Winata, Peng Xu, and Pascale Fung. 2020. Coach: A coarse-to-fine approach for cross-domain slot filling. In *Proc. of the ACL*, pages 19–25.

Bingfeng Luo, Yansong Feng, Zheng Wang, Songfang Huang, Rui Yan, and Dongyan Zhao. 2018. Marrying up regular expressions with neural networks: A case study for spoken language understanding. In *Proc. of the ACL (Volume 1: Long Papers)*, pages 2083–2093, Melbourne, Australia. Association for Computational Linguistics.

Xin Lv, Yuxian Gu, Xu Han, Lei Hou, Juanzi Li, and Zhiyuan Liu. 2019. Adapting meta knowledge graph information for multi-hop reasoning over few-shot relations. In *Proc. of the EMNLP-IJCNLP*, pages 3376–3381, Hong Kong, China. Association for Computational Linguistics.

Erik G. Miller, Nicholas E. Matsakis, and Paul A. Viola. 2000. Learning from one example through shared densities on transforms. In *Proc. of the CVPR*, pages 1464–1471. IEEE Computer Society.

Libo Qin, Wanxiang Che, Yangming Li, Haoyang Wen, and Ting Liu. 2019. A stack-propagation framework with token-level intent detection for spoken language understanding. In *Proc. of the EMNLP-IJCNLP*, pages 2078–2087, Hong Kong, China. Association for Computational Linguistics.

Libo Qin, Xiao Xu, Wanxiang Che, and Ting Liu. 2020. Towards fine-grained transfer: An adaptive graph-interactive framework for joint multiple intent detection and slot filling. In *Proc. of the EMNLP: Findings*, pages 1807–1816.

Nils Reimers and Iryna Gurevych. 2017. Reporting score distributions makes a difference: Performance study of LSTM-networks for sequence tagging. In *Proc. of the EMNLP*, pages 338–348, Copenhagen, Denmark. Association for Computational Linguistics.

Darsh J. Shah, Raghav Gupta, Amir A. Fayazi, and Dilek Hakkani-Tür. 2019. Robust zero-shot cross-domain slot filling with example values. In *Proc. of the ACL*, pages 5484–5490.

Jake Snell, Kevin Swersky, and Richard S. Zemel. 2017. Prototypical networks for few-shot learning. In *Proc. of the NeurIPS*, pages 4077–4087.

Shengli Sun, Qingfeng Sun, Kevin Zhou, and Tengchao Lv. 2019. Hierarchical attention prototypical networks for few-shot text classification. In *Proc. of the EMNLP-IJCNLP*, pages 476–485, Hong Kong, China. Association for Computational Linguistics.

Oriol Vinyals, Charles Blundell, Tim Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. 2016. Matching networks for one shot learning. In *Proc. of the NeurIPS*, pages 3630–3638.

Vladimir Vlasov, Akela Drissner-Schmid, and Alan Nichol. 2018. Few-shot generalization across dialogue tasks. *CoRR*, abs/1811.11707.

Yu Wang, Yilin Shen, and Hongxia Jin. 2018. A bi-model based RNN semantic frame parsing model for intent detection and slot filling. In *Proc. of the NAACL (Short Papers)*, pages 309–314, New Orleans, Louisiana. Association for Computational Linguistics.

Joseph Worsham and Jugal Kalita. 2020. Multi-task learning for natural language processing in the 2020s: where are we going? *Pattern Recognition Letters*.

Di Wu, Liang Ding, Fan Lu, and Jian Xie. 2020. SlotRefine: A fast non-autoregressive model for joint intent detection and slot filling. In *Proc. of the EMNLP*, pages 1932–1937, Online. Association for Computational Linguistics.

Congying Xia, Chenwei Zhang, Xiaohui Yan, Yi Chang, and Philip Yu. 2018. Zero-shot user intent detection via capsule neural networks. In *Proc. of the EMNLP*, pages 3090–3099, Brussels, Belgium. Association for Computational Linguistics.

Leiming Yan, Yuhui Zheng, and Jie Cao. 2018. Few-shot learning for short text classification. *Multimedia Tools and Applications*, pages 1–12.

Zhi-Xiu Ye and Zhen-Hua Ling. 2019. Multi-level matching and aggregation network for few-shot relation classification. In *Proc. of the ACL*, pages 2872–2881, Florence, Italy. Association for Computational Linguistics.

Sung Whan Yoon, Jun Seo, and Jaekyun Moon. 2019. Tapnet: Neural network augmented with task-adaptive projection for few-shot learning. In *Proc. of the 36th ICML*, volume 97 of *Proceedings of Machine Learning Research*, pages 7115–7123. PMLR.

Steve Young, Milica Gašić, Blaise Thomson, and Jason D Williams. 2013. Pomdp-based statistical spoken dialog systems: A review. In *Proc. of the IEEE*, volume 101, pages 1160–1179. IEEE.

Mo Yu, Xiaoxiao Guo, Jinfeng Yi, Shiyu Chang, Saloni Potdar, Yu Cheng, Gerald Tesauro, Haoyu Wang, and Bowen Zhou. 2018. Diverse few-shot text classification with multiple metrics. In *Proc. of the NAACL, Volume 1 (Long Papers)*, pages 1206–1215, New Orleans, Louisiana. Association for Computational Linguistics.

Chenwei Zhang, Yaliang Li, Nan Du, Wei Fan, and Philip Yu. 2019. Joint slot filling and intent detection via capsule neural networks. In *Proc. of the ACL*, pages 5259–5267, Florence, Italy. Association for Computational Linguistics.