Integration of Old and New Knowledge for Generalized Intent Discovery: A Consistency-driven Prototype-Prompting Framework

Xiao Wei[1,2,3], **Xiaobao Wang**[1,2,*], **Ning Zhuang**[1], **Chenyang Wang**[4], **Longbiao Wang**[1,5,*] and **Jianwu Dang**[3]
[1]Tianjin Key Laboratory of Cognitive Computing and Application, College of Intelligence and Computing, Tianjin University, Tianjin, China
[2]Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ), Shenzhen, China
[3]Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China
[4]College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China
[5]Huiyan Technology (Tianjin) Co., Ltd, Tianjin, China
{weixiao, wangxiaobao, zncola, longbiao_wang}@tju.edu.cn, chenyangwang@ieee.org, jw.dang@siat.ac.cn

Abstract

Intent detection aims to identify user intents from natural language inputs, where supervised methods rely heavily on labeled in-domain (IND) data and struggle with out-of-domain (OOD) intents, limiting their practical applicability. Generalized Intent Discovery (GID) addresses this by leveraging unlabeled OOD data to discover new intents without additional annotation. However, existing methods focus solely on clustering unsupervised data while neglecting domain adaptation. Therefore, we propose a consistency-driven prototype-prompting framework for GID from the perspective of integrating old and new knowledge, which includes a prototype-prompting framework for transferring old knowledge from external sources, and a hierarchical consistency constraint for learning new knowledge from target domains. We conducted extensive experiments and the results show that our method significantly outperforms all baseline methods, achieving state-of-the-art results, which strongly demonstrates the effectiveness and generalization of our methods. Our source code is publicly available at https://github.com/smileix/cpp.

1 Introduction

Intent detection is a core task in both Natural Language Understanding (NLU) and Task-Oriented Dialogue (ToD) systems . Its primary goal is to identify the intent or objective of a user from their natural language input. Intent detection is a critical step for dialogue systems to understand user needs and take appropriate actions. In recent years, with the rise of deep learning, data-driven fully supervised methods have achieved significant success. However, these methods largely rely on a substantial amount of in-domain annotated data and

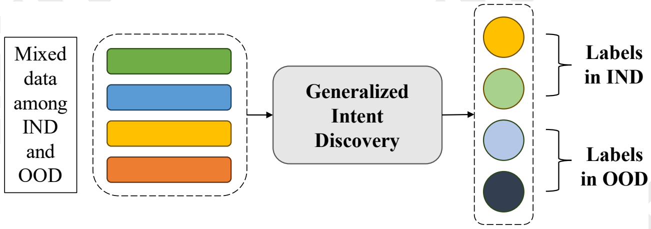

Figure 1: Illustration of the GID task. GID uses labeled IND data and unlabeled OOD data for training, and then jointly classifies IND and OOD data for evaluating.

can only handle a limited set of in-domain intents [Yang *et al.*, 2024], which poses numerous challenges in practical applications. Generally, users will always inquire about some out of scope intents after the deployment of a dialogue system, which traditional dialogue systems may fail to correctly identify and respond to [Siddique *et al.*, 2021].

Therefore, to enhance the generalization, scalability, and adaptability of dialogue systems, it is necessary to study Generalized Intent Discovery (GID) [Mou *et al.*, 2022a; Mou *et al.*, 2023]. As the Fig.1 shows, GID uses labeled in-domain (IND) data and unlabeled out-of-domain (OOD) data for training, which aims to extend the traditional closed-domain intent set by leveraging such OOD data generated by dialogue systems. This reduces reliance on supervised data and gradually expands the system's intent detection scope, enabling it to recognize both known IND intents and unknown OOD intents, thereby improving the adaptability and intelligence of dialogue systems.

The GID task is an emerging research direction in task-oriented dialogue systems, whose core challenge lying in accurately classifying OOD intents in the absence of corresponding supervised data. Current mainstream methods are based on the EM algorithm [Dempster *et al.*, 1977], which uses clustering methods to assign pseudo labels to unsupervised OOD data (E-Step) and then trains a joint classifier with

*Corresponding authors.

supervised IND data (M-Step), repeating this process until the model converges. However, these methods focus on learning new knowledge in target domains while neglecting the transfer of knowledge from external sources [Wang *et al.*, 2022; Wang *et al.*, 2025; Wang *et al.*, 2023]. This is manifested in two aspects. First, previous methods did not fully utilize prior knowledge when generating pseudo labels, resulting in low quality pseudo labels, introducing noise into the joint training phase and limiting the performance of the model. Second, the fine-tuning paradigm is used during model training, which cannot fully eliminate the gap between pretraining tasks and downstream tasks due to insufficient supervised data in low resource scenarios, thus inadequately transferring knowledge from the pre-trained model.

To address these challenges, we propose a **C**onsistency-driven **P**rototype-**P**rompting framework (**CPP**) for GID, from the perspective of integrating old and new knowledge. Our work makes three main contributions:

- A prototype-prompt framework for transferring old knowledge from external sources, including prototype-optimized metric learning and prior-enhanced prompt learning. We exploit large language models (LLMs) to introduce different external knowledge in prototypes and verbalizers, and explore different update methods to improve the utilization of external knowledge.
- Hierarchical consistency constraints for acquiring new knowledge from target domains, including consistency regularization and symmetric cross-prediction loss. We follow and extend the consistency principle to construct different constraints, which can also synergistically conduct metric learning and prompt learning.
- Extensive experiments on up to 9 settings with different domain setups on two datasets, whose results show that our method significantly outperforms all baselines and achieves state-of-the-art results, proving the effectiveness and generalization of our method in generalized intent discovery.

2 Related Work

OOD Intent Detection & Discovery. There are two tasks related to GID that are also designed to address OOD intent. OOD intent detection aims to identify whether a user query belongs to the predefined IND intent set and to reject OOD intents [Lang *et al.*, 2023; Zheng *et al.*, 2020]. However, the task cannot further utilize the potential information in these OOD queries. OOD intent discovery aims to cluster unlabeled OOD data and discover new intent categories [Vedula *et al.*, 2019], iteratively optimizing intent representations and cluster assignments [Lee and others, 2013]. However, it cannot effectively integrate with existing IND intent classifiers. The former is a fully supervised task, whose main methods includes distance-based detection [Xu *et al.*, 2020], generative model-based detection [Xu *et al.*, 2021], and contrastive learning-based detection [Zhou *et al.*, 2022a]. The latter is an unsupervised task, whose main methods includes K-means clustering [Lloyd, 1982], deep alignment-based clustering [Mou *et al.*, 2022b], and contrastive learning-based clustering [Zhou *et al.*, 2022b].

OOD Intent Detection & Discovery. GID is a semi-supervised task, where the current challenge lies in the high coupling between OOD pseudo label generation and representation learning [Sohn *et al.*, 2020; Mou *et al.*, 2023]. Specifically, the quality of pseudo labels affects the performance of subsequent joint representation learning, while the effectiveness of representation learning influences the quality of pseudo label generation in turn.

Existing GID methods can be divided into two categories: pipeline or end-to-end. Pipeline methods are unidirectional, leading to error propagation and thus limiting model performance. Current mainstream methods are based on end-to-end approaches, such as E2E [Mou *et al.*, 2022a] and DPL [Mou *et al.*, 2023]. E2E introduces a framework that combines pseudo label generation and representation learning for joint optimization. Additionally, it employs swap prediction for OOD data to prevent the model from collapsing into degenerate solutions during clustering. Based on the end-to-end framework, DPL further introduces contrastive learning to enhance feature representation learning for OOD data in the target domain. In summary, existing methods focus on optimizing OOD data clustering algorithms while neglecting domain adaption [Farahani *et al.*, 2021] from external sources.

3 Consistency-driven Prototype-Prompting Framework

3.1 Problem Formulation

Input Data. Labeled IND data: $\mathbf{D}_{\text{IND}} = \{(x_i^{\text{IND}}, y_i^{\text{IND}})\}_{i=1}^n$, where $y_i^{\text{IND}} \in \mathcal{Y}_{\text{IND}}$, $\mathcal{Y}_{\text{IND}}$ is known and $|\mathcal{Y}_{\text{IND}}| = N$; unlabeled OOD data : $\mathbf{D}_{\text{OOD}} = \{(x_i^{\text{OOD}})\}_{i=1}^m$, where y_i^{OOD} is unknown, while $\mathcal{Y}_{\text{OOD}}$ is known and $|\mathcal{Y}_{\text{OOD}}| = M$. $\mathcal{Y}_{\text{IND}} \cap \mathcal{Y}_{\text{OOD}} = \emptyset$.

Objective. Use $\mathbf{D}_{\text{IND}}$ and $\mathbf{D}_{\text{OOD}}$ to train a joint classifier to classify input queries into the total label set $\mathcal{Y} = \mathcal{Y}_{\text{IND}} \cup \mathcal{Y}_{\text{OOD}}$.

The model training in this domain is generally divided into two stages: the pretrain stage, where the model performs fully supervised learning on the labeled IND data, and the discover stage, where the model undergoes joint training on the labeled IND data and unlabeled OOD data, and then both IND and OOD data are used for evaluation.

3.2 Overview

The overall framework of our method is illustrated in the Figure 2. For the discover stage, firstly, we design a prototype-prompt framework for transferring old knowledge from external sources, setting up two classifiers: a prototype-optimized similarity comparison module based on metric learning, and a prior-enhanced verbalizer layer based on prompt learning. Secondly, we construct hierarchical consistency constraints for acquiring new knowledge from target domains, and first set up consistency regularization. And then, we extend the idea of consistency principle and construct symmetric swapping prediction loss.

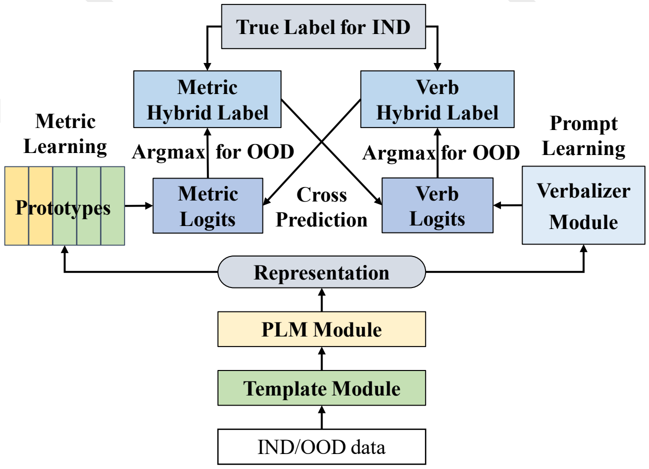

Figure 2: Overall architecture of our proposed method.

3.3 Prototype-Prompting Framework for Old Knowledge Transfer

Prototype-Optimized Metric Learning

Prototype-based metric learning [Kaya and Bilge, 2019; Wei *et al.*, 2022] aims to classify or cluster data by learning a prototype representation for each category and measuring the distance between samples and prototypes, which is particularly effective in low-resource scenarios.

We propose external knowledge-enhanced prototype metric learning based on LLMs, whose core idea is to leverage the semantic understanding capabilities of LLMs to generate high-quality initial prototypes for each category. We experiment with category names, descriptions, representative samples, and keywords.

For prototype computation, specifically, we use the category name as the basic prototype text, denoted as $\mathscr{C} = \{c_1, c_2, \ldots, c_C\}$. We predefine a template and use a LLM (here, DeepSeek-V3) to generate corresponding meta-information denoted as m_i like explanatory descriptions, representative samples, or keywords [Li *et al.*, 2023; Wei *et al.*, 2024]:

$$m_i = \mathrm{LLM}(t(c_i)), \tag{1}$$

where $t(\cdot)$ denote template function, including LLM role definition, dataset background and task description, input and output format description. And then we extract the corresponding embedding using our encoder denoted as $E(\cdot)$ to obtain the prototype for that category. Finally, we concatenate the prototype vectors of all categories to form the overall prototype vector:

$$\mathbf{P} = [E(m_1); E(m_2); \ldots; E(m_C)]. \tag{2}$$

For distance measurement, we compare the cosine similarity between the sample representation and the prototype representation to obtain its logits. For each training step, we repeat Formula (2) to update the prototype representation simply and effectively.

Prior-Enhanced Prompting Framework

Prompt learning aims to reduce the gap between pre-training tasks and downstream tasks, especially in low data scenarios, typically by restructuring downstream tasks into self-supervised tasks of pretrained models [Brown *et al.*, 2020; Liu *et al.*, 2023; Ding *et al.*, 2022]. Based on prompt learning, we predefine a template to reformulate the task into the form of Masked Language Model (MLM) [Devlin *et al.*, 2019] head, guide pretrained models to generate the expected output and design a verbalizer for downstream adaption [Zhuang *et al.*, 2025; Li *et al.*, 2025], thus fully transferring the knowledge of the pretrained models. The framework consists of two main components: template and verbalizer, where we focus on the latter.

For an input text x, we embed it into a template $T(x)$:

$$T(x) = \text{“}x\text{. In this sentence, the intent is about [MASK].”},$$

where '[MASK]' is the position where the model predicts the label word. We use handcrafted templates without any trainable parameters.

The verbalizer acts as a function that maps the model's predicted label words to actual labels. It plays a crucial role in transferring prior knowledge in two ways: label words and MLM head reuse.

The construction of label words is similar to the initial text of prototypes, therefore we reuse the label-related meta-information generated from the LLM in Formula (1) as label words, Let W_i denote the set of label words associated with the category label y_i, and $W_i = m_i$. Assuming each category y_i has k label words, W_i can be also expressed as $W_i = \{w_{i1}, w_{i2}, \ldots, w_{ik}\}$.

We construct a soft verbalizer to reuse the MLM head more efficiently. Concretely, we introduce a downstream linear layer $\mathbf{W}_{\text{verbalizer}} \in \mathbb{R}^{d\times C}$, where d is the embedding dimension and C is the number of categories. The weights of this layer are initialized by a subset of the MLM head's weights. The steps are as follows:

For each category label y_i, we extract the corresponding weights from the MLM head's weight matrix $\mathbf{W}_{\text{MLM}}$ based on its label words W_i:

$$\mathbf{W}_{\text{verbalizer}}[:, i] = \frac{1}{k}\sum_{j=1}^{k} \mathbf{W}_{\text{MLM}}[:, w_{ij}] \tag{3}$$

where $\mathbf{W}_{\text{verbalizer}}[:, i]$ is the weight column for label y_i in the downstream linear layer, and w_{ij} represents the j-th label word for y_i. And then we compute the logits for each label:

$$S_i = \mathbf{W}_{\text{verbalizer}}[:, i]^{\top} h_{\text{[MASK]}}, \tag{4}$$

where $h_{\text{[MASK]}}$ is the hidden representation of the '[MASK]' token. Finally, the logits is normalized to obtain the probability distribution:

$$p(y_i|x) = \frac{\exp(S_i)}{\sum_{j=1}^{C} \exp(S_j)} \tag{5}$$

3.4 Hierarchical Consistency Constraints for New Knowledge Acquisition

In unsupervised learning scenarios, consistency regularization [Sohn *et al.*, 2020], is a crucial technique that leverages the inherent properties of data to train models without relying on labeled data. The core idea is that for reasonable perturbations of the input data, the model's predictions should remain consistent. This allows the model to learn effective feature representations from the intrinsic structure of the data.

Consistency Regularization Loss

We propose a consistency regularization (CR) [Fan *et al.*, 2023] loss based on the principle of consistency, including data consistency and prediction consistency.

As for the data consistency, we generate two different representation views of the same input with the dropout mechanism [Little, 1995], denoted as $\mathbf{H}_1$ and $\mathbf{H}_2$. And then we use symmetric KL divergence [Kullback and Leibler, 1951] to align the distributions of these two views:

$$\mathcal{L}_{\text{dc}} = \frac{1}{2}\left(D_{\text{KL}}(\mathbf{H}_1\|\mathbf{H}_2) + D_{\text{KL}}(\mathbf{H}_2\|\mathbf{H}_1)\right), \tag{6}$$

where $\mathbf{H}_1$ and $\mathbf{H}_2$ are obtained by applying mean pooling to the final hidden states of the encoder.

As for the prediction consistency, the two classifiers, denoted as C_1 and C_2, process the same input and obtain two probability distributions for each view. And then we again use symmetric KL divergence to align these distributions:

$$\mathbf{P}_1 = C_1(\mathbf{H}), \quad \mathbf{P}_2 = C_2(\mathbf{H}), \tag{7}$$

$$\mathcal{L}_{\text{pc}} = \frac{1}{2}\left(D_{\text{KL}}(\mathbf{P}_1\|\mathbf{P}_2) + D_{\text{KL}}(\mathbf{P}_2\|\mathbf{P}_1)\right), \tag{8}$$

where $\mathbf{P}_1$ and $\mathbf{P}_2$ are the probability distributions from the two classifiers. The final consistency regularization loss is the sum of these two losses:

$$\mathcal{L}_{\text{CR}} = \mathcal{L}_{\text{dc}} + \mathcal{L}_{\text{pc}}. \tag{9}$$

Symmetric Cross-Prediction Loss

Since GID is a semi-supervised task, where unsupervised data lacks labels during training. To address this, we propose a symmetric cross-prediction loss, based on the two classifiers we constructed. Specifically, for each view:

1). Take argmax of the logits to generate pseudo labels for OOD data:

$$\hat{Y}_1 = \arg\max(\mathbf{P}_1), \quad \hat{Y}_2 = \arg\max(\mathbf{P}_2), \tag{10}$$

2). Combine these pseudo labels of OOD data with the true labels of IND data to form hybrid labels.

$$\text{Hybrid}_1 = \begin{cases} Y & \text{(IND)} \\ \hat{Y}_2 & \text{(OOD)} \end{cases}, \text{Hybrid}_2 = \begin{cases} Y & \text{(IND)} \\ \hat{Y}_1 & \text{(OOD)} \end{cases}, \tag{11}$$

where the Hybrid_1 and Hybrid_2 are the hybrid labels.

3). Compute the cross-entropy (CE) loss [Shannon, 1948; Goodfellow *et al.*, 2016] with hybrid labels and the other classifier's logits, and then symmetrically calculate and accumulate it:

$$\mathcal{L}_{\text{CP}} = \frac{1}{2}\left(\text{CE}(\mathbf{P}_1, \text{Hybrid}_2) + \text{CE}(\mathbf{P}_2, \text{Hybrid}_1)\right). \tag{12}$$

Multi-View Contrastive Learning

For unsupervised data, we also incorporate a classic contrastive learning approach [Chuang *et al.*, 2020]. Specifically, based on the two views mentioned above, we apply multi-view contrastive learning with the NT-Xent loss [Chen *et al.*, 2020]. The goal is to maximize the similarity of positive pairs (views of the same data) and minimize the similarity of negative pairs (views of different data):

$$\mathcal{L}_{\text{CL}} = -\sum_{i=1}^{2B} \log \frac{\exp(\text{sim}(z_i, \hat{z}_i)/\tau)}{\sum_{k=1}^{2B} \mathbb{I}_{k\neq i} \exp(\text{sim}(z_i, z_k)/\tau)} \tag{13}$$

where z_i and $\hat{z}_i$ are the feature representations of the two views of the same data (positive pair), which are obtained by projecting the averaged hidden states of last layer, $\text{sim}(\cdot)$ represents a similarity function (here cosine similarity), B is the batch size and τ is a temperature parameter.

Our final loss is obtained by adding up the three losses:

$$\mathcal{L} = \mathcal{L}_{\text{CR}} + \mathcal{L}_{\text{CP}} + \mathcal{L}_{\text{CL}}. \tag{14}$$

4 Experimental Results

4.1 Datasets and Settings

We conducted experiments on two classic intent detection datasets: Banking [Casanueva *et al.*, 2020] and CLINC [Larson *et al.*, 2019]. Banking is specifically designed for banking scenarios, containing 77 different banking-related intents with approximately 13,000 samples. CLINC covers 10 domains and includes 150 different intent categories, with approximately 22,500 samples.

We adopted the experimental setups from previous work [Mou *et al.*, 2022a; Mou *et al.*, 2023] and further extended data settings . We followed the setups of GID-SD, GID-CD, and GID-MD, which means single-domain, cross-domain and multi-domain respectively. Specifically, GID-SD refers to extracting a certain proportion of data as OOD unsupervised data on Banking, with intent categories as the smallest partition unit, GID-MD refers to performing the same operation on CLINC, while GID-CD refers to obtain such data on CLINC with domain as the smallest partition unit. The remaining data is used as IND supervised data for all the three setups. The model is required to use both IND data with label and OOD data without labels for joint training, and then is expected to predict both intents of IND and OOD. For each setup, previous work used OOD ratios of 20%, 40%, and 60%. We follow the 60% OOD ratio and expand it to 80% and 90%, make it more challenging and discriminative.

4.2 Baselines

We compared our method with the following baselines:

Kmeans [Lloyd, 1982]: A pipeline approach that first clusters OOD data with k-means to generate pseudo labels and then jointly trains with IND data.

DeepAligned [Zhang *et al.*, 2021]: A pipeline approach that improves upon Kmeans by using the Hungarian algorithm to align cluster centers after clustering, addressing label inconsistency.

Setup	GID-SD-60%					GID-CD-60%					GID-MD-60%				
Metric →	IND	OOD		ALL		IND	OOD		ALL		IND	OOD		ALL	
Model ↓	ACC	ACC	F1	ACC	F1	ACC	ACC	F1	ACC	F1	ACC	ACC	F1	ACC	F1
Kmeans	90.40	51.58	51.96	67.08	66.70	96.44	54.67	53.69	71.38	70.57	95.00	65.11	63.68	77.02	76.09
DA	90.97	59.55	59.51	72.05	71.42	97.33	76.15	74.80	84.62	83.60	97.67	83.38	82.78	89.10	88.52
DA-Mix	80.70	52.66	54.66	63.95	61.92	93.89	75.63	74.29	82.93	81.37	92.59	78.34	79.88	84.05	82.74
E2E	91.77	62.23	62.62	73.93	73.77	**98.67**	80.81	80.60	87.96	87.68	98.11	87.19	87.32	91.51	91.24
DPL	92.66	59.84	60.29	72.91	72.54	98.00	79.78	79.56	87.07	86.78	97.93	86.79	87.01	91.26	91.18
CPP	**93.63**	**75.05**	**74.92**	**82.53**	**81.84**	98.11	**87.41**	**86.72**	**91.69**	**91.17**	**98.44**	**90.96**	**91.02**	**93.96**	**93.73**

Table 1: Performance on the SD (Single Domain), CD (Cross Domain) and MD (Multi Domain) setups with 60% OOD ratio. Here DA stands for DeepAligned, and DA-Mix stands for DeepAligned-Mix. For IND data, accuracy is reported, while for OOD data and overall data, accuracy and weighted-F1 score are reported. Note that the best value in each column is represented in bold, and the suboptimal value is represented by an underline. The same below.

Setup	GID-SD-80%					GID-CD-80%					GID-MD-80%				
Metric →	IND	OOD		ALL		IND	OOD		ALL		IND	OOD		ALL	
Model ↓	ACC	ACC	F1	ACC	F1	ACC	ACC	F1	ACC	F1	ACC	ACC	F1	ACC	F1
E2E	91.17	57.98	57.69	64.42	63.86	96.00	84.39	84.32	86.71	86.53	97.56	74.78	74.88	79.33	79.30
DPL	**91.94**	50.94	50.78	58.90	58.44	**98.67**	82.57	82.54	85.79	85.61	**98.00**	74.11	73.66	78.71	78.33
CPP	91.17	**66.85**	**64.42**	**71.59**	**69.23**	96.67	**86.61**	**86.58**	**88.62**	**88.45**	97.56	**79.61**	**79.18**	**83.20**	**82.89**

Table 2: Performance on the SD (Single Domain), CD (Cross Domain) and MD (Multi Domain) setups with 80% OOD ratio.

DeepAligned-Mix [Mou *et al.*, 2022a]: A variant of DeepAligned that differs mainly in the inference phase, where predictions are made using the classification head of the new classifier.

E2E [Mou *et al.*, 2022a]: An end-to-end approach for pseudo label generation and intent classification, using Sinkhorn-Knopp (SK) [Sinkhorn and Knopp, 1967] algorithm to optimize pseudo label assignment and employing multi-view swap prediction [Chen *et al.*, 2020].

DPL [Mou *et al.*, 2023]: An end-to-end approach that builds on E2E, using prototype contrastive learning to learn and generate pseudo labels for OOD data. It achieved state-of-the-art results on the previous 40% OOD ratio settings.

For fair comparison, all baselines use BERT (bert-base-uncased) [Devlin *et al.*, 2019] as the encoders, and are evaluated in terms of accuracy and weighted F1-score follow previous settings.

4.3 Implementation Details

All text is tokenized with wordpieces, the maximum sequence length is set to 128, and dynamic padding is used; the batch size is set to 64, the maximum training epoch is set to 25, and the early stop is set to 10; the bert-base-uncased pretrained model is loaded as the backbone, and the dropout is set to 0.1; the optimizer is set to AdamW [Loshchilov and Hutter, 2017], the weight decay is 0.01, the initial learning rate is set to 5e-5, and a linear warm-up strategy [Goyal *et al.*, 2017] with 500 warm-up steps is adopted; we utilized prompt-based fine-tuning and set up an additional projection layer with 256 dimensional for NT-Xent loss. To ensure comparability, we directly use the code released in previous studies, especially E2E and DPL. All of the experiments were conducted on 8 NVIDIA RTX 4090D GPUs.

4.4 Main Results

We conducted extensive experiments for the three setups (GID-SD, GID-CD, and GID-MD) with three data settings (60%, 80%, and 90% OOD ratios) . We compared the performance with baselines, and the results are shown in Tables 1, 2, and 3. Following convention, we use 5 evaluation metrics for each experiment, i.e., accuracy of IND data, accuracy and weighted-F1 score of OOD data, and overall accuracy and weighted-F1 score. Note that for the experimental results not provided by the baselines themselves, we reproduced them via their source codes.

The experimental results show that our method (CPP) significantly outperforms the baselines across all settings, achieving state-of-the-art performance. Overall, compared to the suboptimal model and in terms of total accuracy, our method leads by 8.60%, 7.17%, and 9.90%, respectively for the 60%, 80%, and 90% OOD ratios in the SD setup. In the CD setup, these numbers are 3.73%, 3.87%, and 13.11%, and in the MD setup, they are 2.45%, 1.91%, and 6.98%. These results strongly demonstrate the effectiveness and generalization of our method. Further observation reveals that our method achieves the best results on almost all metrics (40/45). The performance of different methods on IND data is roughly comparable, while our improvement on OOD data is more critical. Baseline methods tend to focus on clustering learning for unsupervised data while neglecting domain adaptation. Our improvement is attributed to the old-new knowledge fusion mechanism, which effectively transfers external knowledge and source domain knowledge to target domains while learning new knowledge from unsupervised data through hierarchical consistency constraints.

We conducted experiments on three setups: SD, CD, and MD. The performance of all methods gradually improves

Setup	GID-SD-90%					GID-CD-90%					GID-MD-90%				
Metric →	IND	OOD		ALL		IND	OOD		ALL		IND	OOD		ALL	
Model ↓	ACC	ACC	F1	ACC	F1	ACC	ACC	F1	ACC	F1	ACC	ACC	F1	ACC	F1
E2E	89.69	52.28	51.87	56.17	55.63	96.89	75.95	75.57	78.00	77.53	96.44	53.88	53.50	58.13	57.84
DPL	**90.42**	40.04	39.20	45.14	44.15	96.89	75.85	75.63	77.93	77.66	96.44	67.51	67.14	70.49	70.17
CPP	87.19	**63.62**	**61.21**	**66.07**	**63.67**	**98.22**	**83.51**	**82.73**	**84.98**	**84.20**	**96.44**	**82.17**	**81.63**	**83.60**	**83.13**

Table 3: Performance on the SD (Single Domain), CD (Cross Domain) and MD (Multi Domain) setups with 90% OOD ratio.

Setup	GID-SD-60%				
Metric →	IND	OOD		ALL	
Model ↓	ACC	ACC	F1	ACC	F1
0-Shot DeepSeek	77.42	69.13	73.93	72.47	72.12
0-Shot GPT-4o	72.81	72.90	73.26	72.89	72.10
1-Shot DeepSeek	87.14	72.55	**76.49**	78.51	77.88
1-Shot GPT-4o	83.95	73.37	76.79	77.63	77.56
CPP	**93.63**	**75.05**	74.92	**82.53**	**81.84**

Table 4: Metric on GID-SD-60% for comparison with LLMs.

Setup	GID-SD-60%				
Metric →	IND	OOD		ALL	
Setting ↓	ACC	ACC	F1	ACC	F1
Label name	92.42	58.97	59.49	72.44	72.34
Paraphrase	**93.71**	57.99	55.31	72.37	70.05
Keywords	90.16	74.29	75.29	80.68	80.39
1 example	93.47	71.52	71.85	80.36	79.88
2 examples	93.23	73.59	73.79	81.49	81.03
3 examples	93.55	75.98	76.25	83.05	82.56
4 examples	93.55	**78.64**	**79.61**	**84.64**	**84.56**
5 examples	93.63	75.05	74.92	82.53	81.84

Table 5: Metric on GID-SD-60% with various Meta-information.

across these setups. This is because, in the SD setup, the model needs to recognize multiple fine-grained intents within a single domain, increasing the classification challenge. Compared to the MD setup, the CD setup makes it difficult for the model to transfer shared knowledge from source domains to the target domains due to the cross-domain setting between IND and OOD data. The improvement of our method over the suboptimal methods in these three settings generally follows this trend. For example, for the 60% OOD ratio, our method improves over E2E by 8.60%, 3.73%, and 2.45%, respectively. Additionally, we evaluated three OOD ratios: 60%, 80%, and 90%. The performance of all methods gradually declines as the OOD ratio increases, which aligns with our intuition. The most significant improvement of our method over the suboptimal model occurs at the 90% OOD ratio. The larger the OOD ratio, the more urgent the need for relevant knowledge in GID methods. For example, in the GID-CD-90% setup, our method improves over DPL by 13.11%, demonstrating that even in extremely resource-scarce scenarios, our method can effectively transfer old knowledge from external sources and learn new knowledge from OOD data.

5 Quantitative Analysis

5.1 Comparison with LLMs

In the field of Natural Language Processing (NLP), LLMs have demonstrated powerful performance across a variety of tasks. However, these models typically require substantial computational resources and data, which limits their applicability in resource-constrained scenarios. Our approach integrates both large and small language models (SLMs), leveraging the strengths of LLMs to assist SLMs in decision-making, thereby combining their complementary advantages to achieve low resource consumption and high performance.

For further analysis, we select the recently popular DeepSeek-V3 and the established GPT-4o as baselines, and conduct zero-shot and few-shot experiments. Note that 1-shot refers that for each IND category, one random sample with the corresponding label is used , while for OOD categories, $|OOD * 1|$ samples are randomly chosen from the OOD training data in total, due to the absence of their labels.

As shown in Table 4, our method significantly outperforms these LLMs in terms of overall metrics. The accuracy of our method is higher than that of DeepSeek 1-shot by 4.02% and higher than GPT-4o 1-Shot by 4.90%. In a closer observation, this improvement is primarily attributed to the strong performance on IND data, where our method surpasses DeepSeek 1-shot by 6.49% and GPT-4o 1-Shot by 9.68%. In contrast, the gap narrows significantly on OOD data, where our method maintains a slight lead in accuracy but falls slightly behind in F1 scores. This suggests that our method still have potential for handling of OOD data. Overall, our method demonstrates superior performance on specific tasks compared to LLMs, particularly in resource-constrained scenarios, making it a competitive choice.

5.2 Utilization of Meta-information

In few-shot learning and zero-shot learning scenarios, the performance of SLMs is often limited by their constrained parameter scale and training data. To enhance the performance of small models in complex tasks, an effective approach is to leverage meta-information related to labels generated by LLMs to augment the decision-making capabilities of small models. This method not only utilizes the powerful semantic understanding abilities of large models but also compensates for the shortcomings of small models in data-scarce situations through knowledge transfer. We conduct this method by injecting label-related descriptions, such as paraphrases, keywords, and examples, into the prototype and verbalizer of our

Setup	GID-SD-60%				
Metric →	IND	OOD		ALL	
Setting ↓	ACC	ACC	F1	ACC	F1
Soft Template	92.90	<u>71.09</u>	<u>71.28</u>	<u>79.87</u>	<u>78.98</u>
PLM Freeze v1	66.85	34.35	34.48	47.44	42.11
PLM Freeze v2	<u>93.55</u>	60.16	57.17	73.60	71.00
Manual Verb	60.08	39.67	35.62	47.89	39.52
PTR Verbalizer	59.76	40.16	36.88	48.05	40.01
CPP	**93.63**	**75.05**	**74.92**	**82.53**	**81.84**

Table 6: Metric on GID-SD-60% in various parameter settings.

Setup	GID-SD-60%				
Metric →	IND	OOD		ALL	
Setting ↓	ACC	ACC	F1	ACC	F1
w/o $\mathcal{L}_{dc}$	93.55	73.70	74.11	81.69	80.97
w/o $\mathcal{L}_{pc}$	92.42	<u>74.51</u>	<u>74.11</u>	<u>81.72</u>	<u>80.98</u>
w/o $\mathcal{L}_{CP}$	93.06	70.54	70.52	79.61	78.95
w/o $\mathcal{L}_{CL}$	**94.03**	69.29	67.89	79.25	77.84
CPP	<u>93.63</u>	**75.05**	**74.92**	**82.53**	**81.84**

Table 7: Metric on GID-SD-60% with various loss functions.

SLM. As shown in Table 5, the experimental results demonstrate that this method significantly improves the performance of small models, highlighting its important application value in resource-constrained scenarios.

Specifically, the performance across all configurations on IND data remains relatively stable, indicating that the enhancement effect of meta-information on IND data is limited, with the primary improvement observed on OOD data. When only label names or paraphrases are used, the performance on OOD data is relatively low, suggesting that they provide limited semantic information. Keywords and examples are more effective. In the GID-SD-60% setup, as examples increases, the performance on OOD data improves significantly, reaching its peak when using 4 examples. This suggests that a greater number of examples can provide richer contextual information. However, when examples increases to 5, the performance slightly declines. It is worth noting that we conducted this experiment across all other setups as well, and the conclusions vary slightly across different setups. In 6 out of 9 setups, using 5 examples yields the best performance.

5.3 Ablation Study

Parameter Settings

Prompt learning and parameter-efficient fine-tuning have emerged as highly regarded research directions in recent years. The effectiveness of prompt learning heavily depends on the parameter settings. Therefore, we first conducted parameter update experiments, as shown in Table 6. We systematically explore the impact of different parameter configurations on model performance, through ablation experiments on the parameter settings of our prompt framework, the encoder, and the verbalizer. PLM Freeze v1 refers to freezing all parameters in the pretrained language model (PLM), while PLM Freeze v2 refers to unfreezing its last layer on the basis of v1.

Our method, which combines hard templates, prompt-based fine-tuning, and soft verbalizers, achieves the best performance. Upon closer examination, although completely freezing the PLM parameters leads to a significant performance drop, the model still achieves an accuracy of 47.44%, demonstrating that our framework is highly effective in transferring knowledge from external sources into target domains. Both manually designed and rule-based PTR [Han *et al.*, 2021] verbalizers perform poorly, indicating their lack of flexibility. In contrast, the soft verbalizer dynamically generates label word mappings, better capturing task semantics and significantly improving the model's performance on both IND and OOD data.

Loss Function Settings

To better acquire new knowledge from target domains, we designs three types of loss functions: consistency regularization loss (including data consistency loss and prediction consistency loss), symmetric cross-prediction loss, and contrastive learning loss. Each loss function plays a distinct role in model training. Through ablation experiments, we can clearly evaluate the specific contribution of each loss function to model performance, thereby validating their rationality and necessity. By analyzing the impact of each loss function on model performance, we can further optimize the weights of the loss functions or design more efficient training strategies.

As shown in Table 7, the ablation experiment results demonstrate that each of these loss functions contributes to model performance to varying degrees. Specifically, removing the contrastive learning loss results in the most significant performance drop on OOD data, indicating its irreplaceable role in enhancing the feature representation of unlabeled data. Following that, the symmetric cross-prediction loss also shows its importance in improving the model's classification capability on unlabeled data. Additionally, the data consistency loss plays a certain role in enhancing the quality of representation learning, while the prediction consistency constraint helps reduce the negative impact of pseudo-label noise on the model.

6 Conclusion

In this study, we propose our method from the perspective of integrating old and new knowledge, which includes a prototype-prompt framework for transferring old knowledge from external sources and hierarchical consistency constraints for acquiring new knowledge in target domain. For the former, we introduce LLMs to generate label-related meta-information and apply it into metric learning and prompt learning to reinforce SLMs decisions, thereby achieving complementary advantages of large and small PLMs. For the latter, we synergistically conduct prompt learning and metric learning to further improve performance. We conducted extensive experiments and analysis on various settings to validate the effectiveness and generalization of our method. In the future, we will attempt to load more powerful LLMs to explore parameter-efficient fine-tuning methods in this scenario, further improving its performance under extremely low resource conditions.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 62302333 and No. U23B2053) and the Open Research Fund from Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ) (No.GML-KF-24-16).

References

[Brown *et al.*, 2020] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[Casanueva *et al.*, 2020] Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. Efficient intent detection with dual sentence encoders. *arXiv preprint arXiv:2003.04807*, 2020.

[Chen *et al.*, 2020] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

[Chuang *et al.*, 2020] Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. *Advances in neural information processing systems*, 33:8765–8775, 2020.

[Dempster *et al.*, 1977] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

[Devlin *et al.*, 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, 2019.

[Ding *et al.*, 2022] Ning Ding, Shengding Hu, Weilin Zhao, Yulin Chen, Zhiyuan Liu, Haitao Zheng, and Maosong Sun. Openprompt: An open-source framework for prompt-learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 105–113, 2022.

[Fan *et al.*, 2023] Yue Fan, Anna Kukleva, Dengxin Dai, and Bernt Schiele. Revisiting consistency regularization for semi-supervised learning. *International Journal of Computer Vision*, 131(3):626–643, 2023.

[Farahani *et al.*, 2021] Abolfazl Farahani, Sahar Voghoei, Khaled Rasheed, and Hamid R Arabnia. A brief review of domain adaptation. *Advances in data science and information engineering: proceedings from ICDATA 2020 and IKE 2020*, pages 877–894, 2021.

[Goodfellow *et al.*, 2016] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, Cambridge, MA, 2016.

[Goyal *et al.*, 2017] Priya Goyal, Piotr Dollár, Ross Girshick, et al. Accurate, large minibatch sgd: Training imagenet in 1 hour. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4710–4718, 2017.

[Han *et al.*, 2021] Xu Han, Wayne Zhao, Ning Ding, Zhiyuan Liu, and Maosong Sun. Ptr: Prompt tuning with rules for text classification. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3456–3467, Online, 2021. Association for Computational Linguistics.

[Kaya and Bilge, 2019] Mahmut Kaya and Hasan Şakir Bilge. Deep metric learning: A survey. *Symmetry*, 11(9):1066, 2019.

[Kullback and Leibler, 1951] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 1951.

[Lang *et al.*, 2023] Hao Lang, Yinhe Zheng, Yixuan Li, Jian Sun, Fei Huang, and Yongbin Li. A survey on out-of-distribution detection in nlp. *arXiv preprint arXiv:2305.03236*, 2023.

[Larson *et al.*, 2019] Stefan Larson, Anish Mahendran, Joseph J Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K Kummerfeld, Kevin Leach, Michael A Laurenzano, Lingjia Tang, et al. An evaluation dataset for intent classification and out-of-scope prediction. *arXiv preprint arXiv:1909.02027*, 2019.

[Lee and others, 2013] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*, volume 3, page 896, 2013.

[Li *et al.*, 2023] Yuhang Li, Xiao Wei, Yuke Si, Longbiao Wang, Xiaobao Wang, and Jianwu Dang. Improving zero-shot cross-domain slot filling via transformer-based slot semantics fusion. In *Proc. Interspeech 2023*, pages 2123–2127, 2023.

[Li *et al.*, 2025] Junlei Li, Xiao Wei, Xiaobao Wang, Ning Zhuang, Longbiao Wang, and Jianwu Dang. Language-emphasized cross-lingual in-context learning for multilingual llm. In Derek F. Wong, Zhongyu Wei, and Muyun Yang, editors, *Natural Language Processing and Chinese Computing*, pages 327–339, Singapore, 2025. Springer Nature Singapore.

[Little, 1995] Roderick JA Little. Modeling the drop-out mechanism in repeated-measures studies. *Journal of the american statistical association*, 90(431):1112–1121, 1995.

[Liu *et al.*, 2023] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35, 2023.

[Lloyd, 1982] Stuart Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

[Loshchilov and Hutter, 2017] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[Mou *et al.*, 2022a] Yutao Mou, Keqing He, Yanan Wu, Pei Wang, Jingang Wang, and Weiran Xu. Generalized intent discovery: Learning from open world dialogue system. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1234–1245, 2022.

[Mou *et al.*, 2022b] Yutao Mou, Keqing He, Yanan Wu, Zhiyuan Zeng, Hong Xu, Huixing Jiang, Wei Wu, and Weiran Xu. Disentangled knowledge transfer for ood intent discovery with unified contrastive learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 46–53, 2022.

[Mou *et al.*, 2023] Yutao Mou, Xiaoshuai Song, Keqing He, Chen Zeng, Pei Wang, Jingang Wang, Yunsen Xian, and Weiran Xu. Decoupling pseudo label disambiguation and representation learning for generalized intent discovery. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9661–9675, Toronto, Canada, 2023. Association for Computational Linguistics.

[Shannon, 1948] Claude E Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948.

[Siddique *et al.*, 2021] AB Siddique, Fuad Jamour, and Vagelis Hristidis. Linguistically-enriched and context-awarezero-shot slot filling. In *Proceedings of the Web Conference 2021*, pages 3279–3290, 2021.

[Sinkhorn and Knopp, 1967] Richard Sinkhorn and Paul Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21(2):343–348, 1967.

[Sohn *et al.*, 2020] Kihyuk Sohn, David Berthelot, Nicholas Carlini, et al. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 596–608, 2020.

[Vedula *et al.*, 2019] Nikhita Vedula, Nedim Lipka, Pranav Maneriker, and Srinivasan Parthasarathy. Towards open intent discovery for conversational text. *arXiv preprint arXiv:1904.08524*, 2019.

[Wang *et al.*, 2022] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Tao Qin, Wang Lu, Yiqiang Chen, Wenjun Zeng, and Philip Yu. Generalizing to unseen domains: A survey on domain generalization. *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[Wang *et al.*, 2023] Xiaobao Wang, Yiqi Dong, Di Jin, Yawen Li, Longbiao Wang, and Jianwu Dang. Augmenting affective dependency graph via iterative incongruity graph learning for sarcasm detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(4):4702–4710, Jun. 2023.

[Wang *et al.*, 2025] Xiaobao Wang, Yujing Wang, Dongxiao He, Zhe Yu, Yawen Li, Longbiao Wang, Jianwu Dang, and Di Jin. Elevating knowledge-enhanced entity and relationship understanding for sarcasm detection. *IEEE Transactions on Knowledge and Data Engineering*, 37(6):3356–3371, 2025.

[Wei *et al.*, 2022] Xiao Wei, Yuke Si, Shiquan Wang, Longbiao Wang, and Jianwu Dang. Hierarchical Tagger with Multi-task Learning for Cross-domain Slot Filling. In *Proc. Interspeech 2022*, pages 3273–3277, 2022.

[Wei *et al.*, 2024] Xiao Wei, Yuhang Li, Yuke Si, Longbiao Wang, Xiaobao Wang, and Jianwu Dang. A prompt-based hierarchical pipeline for cross-domain slot filling. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2024.

[Xu *et al.*, 2020] Pengfei Xu, Sayna Ebrahimi, and Subhro Roy. Distance-based out-of-distribution detection with confidence scores. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4248–4257, 2020.

[Xu *et al.*, 2021] Pengfei Xu, Sayna Ebrahimi, and Subhro Roy. Generative model-based out-of-distribution detection. *arXiv preprint arXiv:2106.04321*, 2021.

[Yang *et al.*, 2024] Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *International Journal of Computer Vision*, 132(12):5635–5662, 2024.

[Zhang *et al.*, 2021] Hanlei Zhang, Hua Xu, Ting-En Lin, and Rui Lyu. Discovering new intents with deep aligned clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14365–14373, 2021.

[Zheng *et al.*, 2020] Yinhe Zheng, Guanyi Chen, and Minlie Huang. Out-of-domain detection for natural language understanding in dialog systems. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:1198–1209, 2020.

[Zhou *et al.*, 2022a] Jinghui Zhou, Tianyi Zhang, Jiaxin Huang, et al. Knn-contrastive learning for out-of-distribution intent classification. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1234–1245, 2022.

[Zhou *et al.*, 2022b] Jinghui Zhou, Tianyi Zhang, Jiaxin Huang, et al. Knn-contrastive learning for out-of-domain intent discovery. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1234–1245, 2022.

[Zhuang *et al.*, 2025] Ning Zhuang, Xiao Wei, Junlei Li, Xiaobao Wang, Chenyang Wang, Longbiao Wang, and Jianwu Dang. A prompt learning framework with large language model augmentation for few-shot multi-label intent detection. In *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2025.