Learning Few-shot Sample-set Operations for Noisy Multi-label Aspect Category Detection

ShimanZhao[1,2] , **WeiChen***[1,2,3] and **TengjiaoWang**[1,2,3]
[1] Key Lab of High Confidence Software Technologies (MOE), School of Computer Science, Peking University, Beijing, China
[2] Research Center for Computational Social Science, Peking University
[3] Institute of Computational Social Science, Peking University (Qingdao)
shimanzhao@stu.pku.edu.cn, {pekingchenwei, tjwang}@pku.edu.cn

Abstract

Multi-label Aspect Category Detection (MACD) is essential for aspect-based sentiment analysis, which aims to identify multiple aspect categories in a given sentence. Few-shot MACD is critical due to the scarcity of labeled data. However, MACD is a high-noise task, and existing methods fail to address it with only two or three training samples per class, which limits the application in practice. To solve above issues, we propose a group of Few-shot Sample-set Operations (FSO) to solve noisy MACD in fewer sample scenarios by identifying the semantic contents of samples. Learning interactions among intersection, subtraction, and union networks, the FSO imitates arithmetic operations on samples to distinguish relevant and irrelevant aspect contents. Eliminating the negative effect caused by noises, the FSO extracts discriminative prototypes and customizes a dedicated query vector for each class. Besides, we develop a multi-label architecture, which integrates with score-wise loss and multi-label loss to optimize the FSO for multi-label prediction, avoiding complex threshold training or selection. Experiments show that our method achieves considerable performance. Significantly, it improves by 11.01% at most and an average of 8.59% Macro-F in fewer sample scenarios.

1 Introduction

Multi-label Aspect Category Detection (MACD) [Tulkens *et al.*, 2020] is a crucial task for aspect-based sentiment analysis [Pontiki *et al.*, 2016], which aims to identify aspect categories in a given sentence. Generally, a sentence contains more than one aspect category, i.e., it can be viewed as a multi-label classification problem. Last few years, MACD received widespread attention. However, most existing methods [Li *et al.*, 2020] heavily rely on a considerable amount of labeled data during training. Therefore, their performances can drop dramatically when only a few labeled samples exist for some aspect categories. Intuitively, few-shot learning is of great significance in solving MACD.

Support Set	
(A) Service	(1) Nice service but food cost a lot. (2) You should ask server well in advance about the internet connection.
(B) Money	(1) There was a charge for wifi. (2) High rates for just ok room service but the internet seems to only work at times.
(C) Internet	(1) There was a charge for wifi. (2) You should ask server well in advance about the internet connection.
Query Set	
(A), (B), (C)	(1) I access the ethernet cable at the corner due free service.
(B), (C)	(2) The hotel is everything I expected:great price, good breakfast, free wifi.
(A)	(3) Our server thomas made the experience that much more enjoyable.

Table 1: A 3-way 2-shot example. The colored boxes indicate target aspects, while the gray boxes highlight irrelevant aspects.

However, most few-shot learning methods (e.g., prototypical network) focus on single-label prediction, i.e., each sentence is restricted to one label. Therefore, they are volatile for the MACD task in the noisy scenario. The main challenges are summarized as follows: (1) Each class prototype is closely related to the target aspects of intra-class support samples, whereas irrelevant aspects interfere with prototype extraction. For example, in Table 1, for the class "Service", its corresponding prototype may receive negative impacts from irrelevant aspects "food", "cost a lot", and "internet". (2) Under the multi-label setting, some support samples are shared among classes, failing to distinguish the class prototypes. Table 1 shows that the sample "There was a charge for wifi" is shared by the classes "Money" and "Internet". For these two classes, the prototypes may be indistinguishable due to the high-similar sample distribution. Inevitably, the above issues complicate class prototype extraction. (3) The number of aspect categories in a sentence is intangible for multi-label prediction. Therefore, a low-cost and high-efficiency method is expected to address the noisy MACD task.

*Corresponding author

Existing works focus on contrastive learning paradigms or attention networks to alleviate noise interference for the MACD task. However, merely learning prototypes by training contrastive objects [Liu *et al.*, 2022; Zhao *et al.*, 2022] or attention weights [Hu *et al.*, 2021] fails to fully address the noise caused by the irrelevant aspects and similar sample distribution. Therefore, they struggle to handle the noisy MACD task in fewer sample scenarios. Shortly, their performances drop significantly when each class only has two or three training samples. Therefore, it remains a considerable challenge for the noisy MACD task.

To solve the above issues, we revisit the MACD task from a new perspective and propose a group of Few-shot Sample-set Operations (FSO) to handle the noisy MACD task by identifying the semantic contents of samples. The set operations are realized as the arithmetic operations on samples and gain considerable progress in image synthesis [Alfassy *et al.*, 2019]. Therefore, we apply the concept of set operations to the prototypical network and design FSO to handle noisy MACD in fewer sample scenarios. The FSO imitates arithmetic operations through intersection (M_{int}), subtraction (M_{sub}), and union (M_{uni}) networks. The M_{int} receives two samples and produces a feature vector with their common semantic content, excluding the other aspects in the original samples. Inversely, the feature vector generated by M_{sub} removes the shared content and reserves the irrelevant aspects. Besides, the M_{uni} is implemented on the outputs of M_{int} and M_{sub} to restore the original samples. Learning the interactions among the M_{int}, M_{sub}, and M_{uni}, the FSO analyzes the semantic contents of the sample and distinguishes irrelevant aspects from it. For support set, the FSO utilizes the shared features within a class to extract discriminative prototypes. For query set, the FSO takes category description as prior knowledge to customize a dedicated query vector for each class. To meet actual practice, we apply the FSO to a multi-label architecture. And score-wise loss and multi-label loss are implemented on the architecture to promote the learning of the FSO and throw the trouble of threshold setting. The contributions are summarized as follows:

- We propose the FSO to solve the noisy MACD task by distinguishing the semantic contents of samples. Learning the interactions among M_{int}, M_{sub}, and M_{uni}, our method alleviates the noises to generate discriminative prototypes for support set and dedicated query vectors for query set to estimate label-sample relevance. And the FSO works well with fewer sample scenarios.

- We design a multi-label architecture that integrates with score-wise loss and multi-label loss to optimize the FSO for multi-label prediction, avoiding complex threshold training and selection.

- Extensive experiments show that our method outperforms strong baselines. Besides, the method is not limited to the MACD task. It can also be applied to more complex tasks, e.g., sentence embedding representation, since it better separates embedding features than conventional contrastive learning in fewer sample scenarios.

2 Related Work

2.1 Multi-label Aspect Category Detection

MACD is a subtask of aspect-based sentiment analysis, which aims to identify aspect categories from a predefined set. The previous works can be summarized into two groups: supervised and unsupervised methods. Supervised methods [Schmitt *et al.*, 2018; Cai *et al.*, 2020] heavily rely on a large amount of labeled data to learn features for each aspect category. Therefore, they suffer from the long-tail distribution [Yu *et al.*, 2021] for some aspect categories with a few labeled data. Unsupervised methods [Tulkens *et al.*, 2020] are poorly performed by mining aspect knowledge in massive unstructured texts. Therefore, recent works keep an eye on few-shot learning.

2.2 Multi-label Few-shot Learning

The meta-learning [Hospedales *et al.*, 2021] is a mainstream few-shot learning line, including model-based [Tsendsuren and Hong, 2017], optimization-based [Lee *et al.*, 2019], and metric-based [Sung *et al.*, 2018; Assran *et al.*, 2022; Wang *et al.*, 2021; Lv *et al.*, 2021] methods. However, most of them only work well in the single-label setting and fail to address high-noise multi-label tasks. To the best of our knowledge, the research works on multi-label few-shot learning mostly focus on contrastive learning paradigms and attention networks in the text domain. Yang et al. [2020] utilize contrastive learning to push positive and negative samples away from each other. However, they have limited performances due to the neglect of adverse effects produced by irrelevant aspects. Then, Hu et al. [2021] and Yan et al. [2022] leverage attention networks to alleviate the noise from irrelevant aspects. However, they are inefficient when many high-similar samples exist in different classes. From a new perspective, we design the FSO to solve the above issues. The concept of set operations is proposed by Alfassy et al [Alfassy *et al.*, 2019], and they use label-set operations to generate more data on image synthesis. Compared with them, we design novel sample-set operations without data generation and introduce score-wise loss and multi-label loss to improve performances on the MACD task.

3 Methodology

3.1 Problem Formulation

We follow the episodic paradigm to train a meta-learner for the noisy MACD task. In the label space, the data can be divided into $\mathcal{C}_{train}$ (known) and $\mathcal{C}_{test}$ (unknown), where $\mathcal{C}_{train} \cap \mathcal{C}_{test} = \emptyset$. And a meta-task includes support set $\mathcal{S}$ and query set $\mathcal{Q}$. In the meta-train, N unique classes are sampled from $\mathcal{C}_{train}$, and then K samples are sampled from each class to construct $\mathcal{S}$ (i.e., N-way K-shot formulation), which can be denoted as $\mathcal{S} = \{(x_1^1, y_1^1), ..., (x_1^K, y_1^K), ..., (x_N^K, y_N^K)\}$. And $\mathcal{Q}$ includes T samples sampled from the remaining samples of the same N classes, i.e., $\mathcal{Q} = \{(x_1^1, y_1^1), ..., (x_1^T, y_1^T), ..., (x_N^T, y_N^T)\}$. In the meta-test, we need to construct the support set and query set from $\mathcal{C}_{test}$. The meta-learner aims to predict the class label of the query set based on the support set.

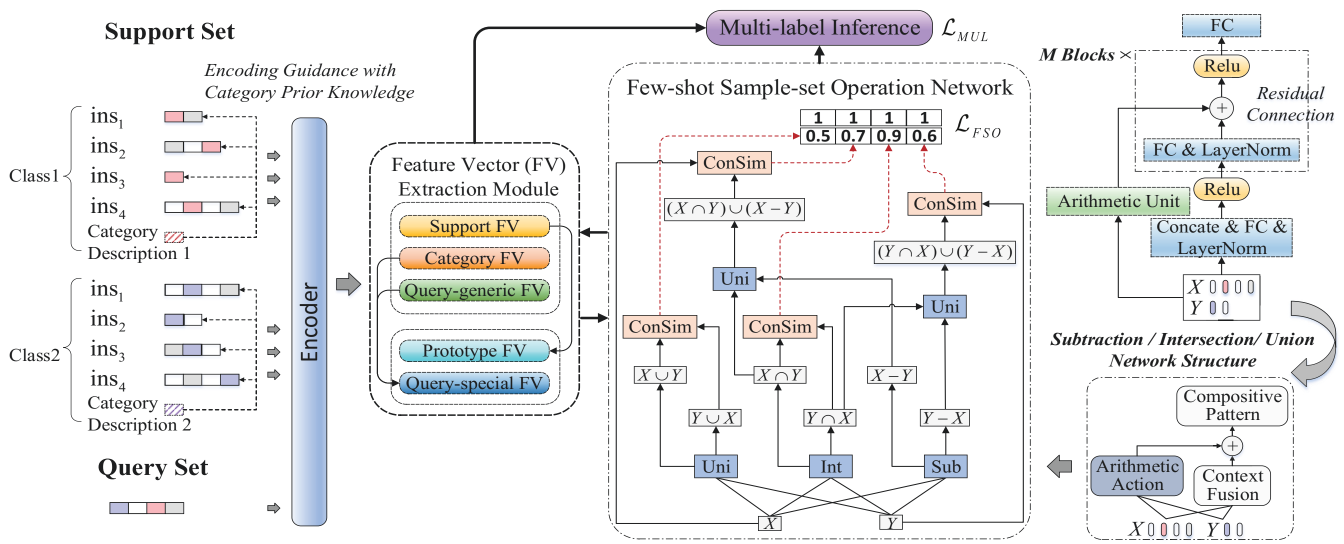


Figure 1: The overall architecture of the proposed method. The left part depicts the support and query samples, where the colored cubes indicate the target aspects, and the gray cubes show the noise caused by irrelevant aspects. The middle part depicts the training details of the FSO. The right part presents the network structure of intersection, subtraction, and union but the arithmetic unit is different.

3.2 Architecture Overview

The proposed architecture is shown in Figure 1. Learning the interactions among the M_{int}, M_{sub}, and M_{uni}, the FSO imitates arithmetic operations to analyze the semantic contents of samples. We first introduce the learning of the FSO and then apply it to support and query sets for multi-label prediction.

3.3 Few-shot Sample-set Operations

The FSO includes three operations, i.e., M_{int}, M_{sub}, and M_{uni}. These three operations refer to mathematical symbols (e.g., $\cap$, $-$, and $\cup$). Given two feature vectors X and Y, the M_{int} extracts the shared features between them, the M_{sub} identifies the exclusive features of X excluding Y, and the M_{uni} denotes the merged features from them. To ensure effects, the M_{int}, M_{sub}, and M_{uni} networks map the synthesized feature vectors of X and Y to feature space $\mathcal{F}$:

$$M_{int}(X,Y) = F_{int}^{XY} \in \mathcal{F}, \tag{1}$$

$$M_{sub}(X,Y) = F_{sub}^{XY} \in \mathcal{F}, \tag{2}$$

$$M_{uni}(X,Y) = F_{uni}^{XY} \in \mathcal{F}, \tag{3}$$

The M_{int}, M_{sub}, and M_{uni} follow the same network structure and different arithmetic units (e.g., $min(.)$, $sub(.)$, and $max(.)$):

$$H_1 = Relu(LayerNorm(W_1[X;Y] + b_1)), \tag{4}$$

$$H_2 = AU(X,Y) + LayerNorm(W_2 H_1 + b_2), \tag{5}$$

$$H_3 = W_3 Relu(H_2) + b_3, \tag{6}$$

where $Relu(.)$ is activation function. $[X;Y]$ denotes the concatenation between X and Y. W_1, W_2, W_3, b_1, b_2, and b_3 are learnable parameters. The $AU(.)$ denotes the arithmetic unit.

Score-wise Loss

The score-wise loss is used to ensure that the M_{int}, M_{sub}, and M_{uni} can capture correct semantic features on samples. Specifically, the score-wise loss is based on cosine similarity scores. And the cosine similarity [Yan *et al.*, 2022] is widely adopted in the prototypical network to measure the similarity scores between feature vectors in the feature space $\mathcal{F}$:

$$ConSim(X,Y) = \frac{X^T Y}{||X||_2 ||Y||_2}, \tag{7}$$

where X^T is transpose of X and $||.||$ is L2-norm of vectors.

The cosine similarity scores range between -1 to 1. Therefore, we use the highest score (i.e., 1) to design the score-wise loss $\mathcal{L}_{FSO}$. The FSO utilizes the score-wise loss $\mathcal{L}_{FSO}$ to enforce the capacity to capture semantic features on samples. The $\mathcal{L}_{FSO}$ includes $\mathcal{L}_{FSO}^{sym}$ and $\mathcal{L}_{FSO}^{norm}$ to ensure the symmetry and normalization of the M_{int}, M_{sub}, and M_{uni}:

$$\mathcal{L}_{FSO} = \mathcal{L}_{FSO}^{sym} + \mathcal{L}_{FSO}^{norm}. \tag{8}$$

The following loss $\mathcal{L}_{FSO}^{sym}$ is used to rectify symmetric M_{int} and M_{uni}:

$$\begin{aligned}\mathcal{L}_{FSO}^{sym} = \sum ((&ConSim(F_{uni}^{XY}, F_{uni}^{YX}) - 1)^2 \\ &+ (ConSim(F_{int}^{XY}, F_{int}^{YX}) - 1)^2),\end{aligned} \tag{9}$$

where F_*^{XY} and F_*^{YX} derive from $M_*(.)$ with reversed order of the inputs, and $* \in (uni, int)$.

The second loss $\mathcal{L}_{FSO}^{norm}$ is realized as a criterion to normalize the M_{int}, M_{sub}, and M_{uni}:

$$\begin{aligned}\mathcal{L}_{FSO}^{norm} = \sum ((&ConSim(M_{uni}(F_{sub}^{XY}, F_{int}^{XY}), X) - 1)^2 \\ &+ (ConSim(M_{uni}(F_{sub}^{YX}, F_{int}^{YX}), Y) - 1)^2).\end{aligned} \tag{10}$$

3.4 Support-set Processing (SP)

Each class in the support set includes K samples, describing the common aspect category. However, the noises complicate the prototype extraction. To alleviate noise interference, we take category description (i.e., label description) as prior knowledge to learn sample representations with category features. Then, the class prototypes are extracted by the FSO through the shared features within a class to further eliminate the negative effect caused by noises.

Support Sample Feature Extraction

Given a support sample $s = \{s_1, s_2, ..., s_n\}$ of n tokens and its category description c $= \{c_1, ..., c_m\}$ of m tokens, we use special tokens [*CLS*] and [*SEP*] to concatenate the sample and its category description. The "$[CLS], s, [SEP], c, [SEP]$" as input is converted to the encoder (e.g., BERT [Devlin *et al.*, 2019]) to generate hidden states H_s:

$$H_s = [h_{CLS}, h_s, h_{SEP}, h_c, h_{SEP}], \tag{11}$$

where $h_s \in R^{n*d}$ denotes the hidden states of the support sample with category features, and $h_c \in R^{m*d}$ indicates the hidden states of the category description. And d is the hidden dimension. Therefore, we define ${h^j_i}_s$ as the hidden states of j^{th} support sample in i^{th} class.

The sample feature vector f^j_i and the corresponding category feature vector g^j_i for j^{th} sample in i^{th} class are obtained through a mean pooling layer:

$$f^j_i = MeanPoolLayer({h^j_i}_s), \tag{12}$$

$$g^j_i = MeanPoolLayer({h^j_i}_c), \tag{13}$$

where $f^j_i \in R^{1*d}$, $g^j_i \in R^{1*d}$, ${h^j_i}_s \in R^{n*d}$, and ${h^j_i}_c \in R^{m*d}$.

For i^{th} class, there are K category feature vectors (i.e., $g^1_i, g^2_i, ..., g^K_i$), and they express the common class information. We assign importance weights to them to map a final category feature vector g_i:

$$A_1 = softmax(W_5 tanh(W_4 g'_i)), \tag{14}$$

$$g_i = A^T_1 g'_i, \tag{15}$$

where A_1 is the weight matrix. $g'_i = [g^1_i, g^2_i, ..., g^K_i] \in R^{K*d}$ and $g_i \in R^{1*d}$. W_4 and W_5 are learnable parameters.

After processing the support set, we obtain $N * K$ sample feature vectors $\{f^1_1, ..., f^K_1, ..., f^1_N, ..., f^K_N\}$ and N category feature vectors $\{g_1, g_2, ..., g_N\}$.

Class Prototype Generation

In the same class, all samples express the common aspect content. The FSO utilizes the shared content within a class to extract the corresponding prototype. Specifically, the FSO receives the sample feature vectors $[f^1_i, f^2_i, ..., f^K_i]$ from i^{th} class. And then, it outputs the intersection features as the common aspect features from the same class by learning the interaction among M_{int}, M_{sub}, and M_{uni} networks:

$$F^i_{int} = FSO([f^1_i, f^2_i, ..., f^K_i]), \tag{16}$$

where $F^i_{int} \in R^{r*d}$. And r is the number of intersection feature vectors extracted by the FSO for i^{th} class.

The discriminative prototype p_i for i^{th} class can be extracted by assigning importance weights to the F^i_{int}:

$$A_2 = softmax(W_7 tanh(W_6 F^i_{int})), \tag{17}$$

$$p_i = A^T_2 F^i_{int}, \tag{18}$$

where A_2 is the weight matrix. $p_i \in R^{1*d}$. W_6 and W_7 are learnable parameters. The SP component can extract N class prototypes, i.e., p_i, $i \in \{1, 2, ..., N\}$.

3.5 Query-set Processing (QP)

The query sample may contain more than one target aspect category in noisy scenarios. Therefore, it is non-trivial to customize a dedicated query vector for each class.

Query Sample Feature Extraction

Given a query sample $q = \{q_1, q_2, ..., q_t\}$ of t tokens, we feed "$[CLS], q, [SEP]$" into the encoder to get hidden states H_q:

$$H_q = [h_{CLS}, h_q, h_{SEP}], \tag{19}$$

where $h_q \in R^{t*d}$ is the hidden states of the query sample.

The query-generic feature vector $\bar{u}$ is obtained through a mean pooling layer.

$$\bar{u} = MeanPoolLayer(h_q), \tag{20}$$

where $\bar{u} \in R^{1*d}$.

Dedicated Query Feature Learning

The FSO takes category description as prior knowledge and extracts target-related features to customize a dedicated query vector for each class. Specifically, we feed the query-generic feature vector $\bar{u}$ and the corresponding category feature vector g_i into the FSO, then it outputs the intersection feature as the query-special feature:

$$u_i = FSO([\bar{u}, g_i]), \tag{21}$$

where $u_i \in R^{1*d}$. The query-special feature is realized as a dedicated query feature vector for each class. Therefore, the QP component can obtain N dedicated query feature vectors, i.e., u_i, $i \in \{1, 2, ..., N\}$.

Distance Metric

Given a query sample, the distance similarities between it and class prototypes are defined as $Z = \{z_1, z_2, ..., z_N\} \in R^N$:

$$z_i = ConSim(p_i, u_i), \tag{22}$$

where $z_i \in R^1$, $i = \{1, 2, ..., N\}$.

3.6 Multi-label Inference

In multi-label inference, some works [Hu *et al.*, 2021; Liu *et al.*, 2022] train a policy network or multi-layer perception to learn a threshold to determine the number of aspects. However, they may fail to get an acceptable threshold to meet the requirement of complicated multi-label prediction. Therefore, we introduce a multi-label loss to optimize training objectives and avoid complex threshold training or selection.

The label of the query sample is $y = \{y_1, y_2, ..., y_N\} \in R^N$, where $y_i \in \{1, 0\}$, and $y_i = 1$ indicates it belongs to i^{th} class. The positive score set $\Gamma = \{z_i \in Z | y_i = 1\}$, and the negative score set $\Lambda = \{z_i \in Z | y_i = 0\}$.

Models	5-way 5-shot		5-way 10-shot		10-way 5-shot		10-way 10-shot	
	AUC	F1	AUC	F1	AUC	F1	AUC	F1
Prototypical Network [Snell *et al.*, 2017]	88.88	66.96	91.77	73.27	87.35	52.06	90.13	59.03
IMP [Allen *et al.*, 2019]	89.95	68.96	92.30	74.13	88.50	54.14	90.81	59.84
Proto-HATT [Gao *et al.*, 2019]	91.54	70.26	93.43	75.24	90.63	57.26	92.86	61.51
Proto-AWATT [Hu *et al.*, 2021]	93.35	75.37	95.28	80.16	92.06	65.65	93.42	69.70
LDF [Liu *et al.*, 2022]	94.65	78.27	95.71	81.87	92.74	67.13	94.29	71.97
LPN [Zhao *et al.*, 2022]	96.45	82.22	97.15	84.90	95.36	71.42	**96.55**	76.51
FSO(ours)	**96.92**	**83.44**	**97.38**	**85.08**	**95.65**	**73.78**	96.28	**76.58**

Table 2: Comparison of AUC and Macro-F1 score on FewAsp (random).

Models	5-way 5-shot		5-way 10-shot		10-way 5-shot		10-way 10-shot	
	AUC	F1	AUC	F1	AUC	F1	AUC	F1
Prototypical Network [Snell *et al.*, 2017]	89.67	67.88	91.60	72.32	88.01	52.72	90.68	58.92
IMP [Allen *et al.*, 2019]	90.12	68.86	92.29	73.51	88.71	53.96	91.10	59.86
Proto-HATT [Gao *et al.*, 2019]	91.10	69.15	93.03	73.91	90.44	55.34	92.38	60.21
Proto-AWATT [Hu *et al.*, 2021]	91.45	71.72	93.89	77.19	89.80	58.89	92.34	66.76
LDF [Liu *et al.*, 2022]	92.62	73.38	94.34	78.81	90.87	62.06	92.93	68.23
LPN [Zhao *et al.*, 2022]	95.66	79.48	96.55	**82.81**	94.51	67.28	95.66	71.87
FSO (ours)	**96.01**	**81.04**	**96.67**	82.22	**94.93**	**70.26**	**95.71**	**72.46**

Table 3: Comparison of AUC and Macro-F1 score on FewAsp (multi).

Multi-label Loss

The multi-label loss based on circle loss [Sun *et al.*, 2020] is written as follows:

$$\begin{aligned}\mathcal{L}_{MUL} &= \log(1 + \sum_{i\in\Lambda, j\in\Gamma} e^{\sigma(z_i - z_j)} + \sum_{i\in\Lambda} e^{\sigma(z_i - t)} + \sum_{j\in\Gamma} e^{\sigma(\mathrm{t} - z_j)}) \\ &= \log(e^{\sigma(t)} + \sum_{i\in\Lambda} e^{\sigma(z_i)}) + \log(e^{\sigma(-t)} + \sum_{j\in\Gamma} e^{\sigma(-z_j)}),\end{aligned} \tag{23}$$

where t is the threshold, and σ is the temperature scale parameter. The optimal goal of the $\mathcal{L}_{MUL}$ is that the target scores are greater than t and the non-target scores are less than t. Directly, we set threshold $t = 0$ and filter positive scores as the multi-label prediction:

$$\mathcal{L}_{MUL} = \log(1 + \sum_{i\in\Lambda} e^{\sigma(z_i)}) + \log(1 + \sum_{j\in\Gamma} e^{\sigma(-z_j)}). \tag{24}$$

Overall Training Objectives

$$\mathcal{L} = \alpha\mathcal{L}_{MUL} + \beta\mathcal{L}_{FSO}, \tag{25}$$

where α and β are hyper-parameters.

4 Experiments

4.1 Experimental Setup

Datasets. Extensive experiments are conducted on datasets FewAsp (multi) and FewAsp (random). These two datasets originate from a large-scale multi-domain dataset (i.e., Yelp aspect [Bauman *et al.*, 2017]) for aspect recommendation. And FewAsp (multi) consists of multi-aspect sentences, whereas FewAsp (random) contains single- and multi-aspect sentences due to random sampling. These two datasets include 100 aspect categories, and we split the 100 aspect categories into 64, 16, and 20 for training, validation, and testing. The detailed statistics are presented in Table 4.

Datasets	FewAsp (multi)			FewAsp (random)		
	train	dev	test	train	dev	test
#cls	64	16	20	64	16	20
#ins	25600	6400	8000	40320	10080	12600
#ins/cls	400	400	400	630	630	630

Table 4: Statistics of two datasets. #cls and #ins indicate the number of classes and samples. And #ins/cls indicates the number of samples for each class.

Implementation Details. The proposed method is implemented with PyTorch (version 1.10.0). The uncased English version of BERT is our encoder. Besides, the first six layers of BERT are frozen to reduce the trainable parameters. We conduct experiments on a single GPU (RTX 3090 Ti) with CUDA version 11.3. The model is trained by the AdamW optimizer. For the learning rate, we set 5e-4 in the FSO and 1e-4 in other network structures. Meanwhile, we use the GradualWarmupScheduler to optimize the learning rate. And we fix the hyper-parameters σ, α, and β as 0.03, 0.3, and 0.7. (Note: the parameters are adjustable). We randomly sample 100 meta-tasks for training and 600 meta-tasks for validation and testing in every epoch.

Evaluation Metric. We follow Zhao et al. [2022] to use Area Under Curve (AUC) and Macro-F1 score for performance evaluation and comparison.

Models	5-way 2-shot		5-way 3-shot		10-way 2-shot		10-way 3-shot	
	AUC	F1	AUC	F1	AUC	F1	AUC	F1
LDF [Liu *et al.*, 2022]	91.30	70.57	91.23	70.51	89.69	57.45	90.16	59.67
LPN [Zhao *et al.*, 2022]	92.90	70.69	94.21	74.55	91.36	56.11	92.67	60.34
FSO (ours)	**95.14**	**77.44**	**95.27**	**78.58**	**93.97**	**66.41**	**94.32**	**68.28**

Table 5: Comparison of AUC and Macro-F1 score for fewer sample scenarios on FewAsp (multi).

Models	15-way 2-shot		15-way 3-shot		20-way 2-shot		20-way 3-shot	
	AUC	F1	AUC	F1	AUC	F1	AUC	F1
LDF [Liu *et al.*, 2022]	88.30	40.98	89.25	40.07	86.71	31.35	88.42	35.28
LPN [Zhao *et al.*, 2022]	90.88	48.24	92.21	53.38	90.42	44.14	92.20	49.33
FSO (ours)	**93.30**	**59.25**	**93.69**	**62.85**	**92.39**	**54.11**	**93.47**	**58.59**

Table 6: Comparison of AUC and Macro-F1 score for fewer sample scenarios on FewAsp (multi).

4.2 Overall Performance

Following the previous works [Liu *et al.*, 2022], the number of query samples is fixed at 5, and the experiments are conducted on "$N = 5, 10$ and $K = 5, 10$" (i.e., N-way K-shot formulation) to compare the performance with the strong baselines. Then, we set "$N = 5, 10, 15, 20$ and $K = 2, 3$" to further analyze the performance in fewer sample scenarios. The results are presented in Tables 2, 3, 5, and 6, with the following observations.

(1) Overall, the proposed method outperforms most baselines. The results demonstrate the effectiveness of our method on the noisy MACD task. Though the strong baseline (i.e., LPN [Zhao *et al.*, 2022]) achieves competitive Macro-F1 or AUC for the 10-shot scenario in Tables 2 and 3, the number looks large and takes high training time, which limits the applications in practice. The proposed method achieves significant performance in fewer sample scenarios. Specifically, our method improves upon the most competitive baseline by 1.06%-2.61% AUC and 4.03%-10.30% Macro-F in Table 5. Besides, it obtains 1.27%-2.42% AUC and 9.26%-11.01% Macro-F improvements in Table 6. The reason is that the contrastive learning and attention networks are weak in fewer sample scenarios (i.e., 2-shot or 3-shot). We can solve the noisy MACD task in fewer training samples by distinguishing the semantic contents of samples. In short, all experimental results on benchmark datasets show that our method achieves considerable performance.

(2) The results on FewAsp (multi) are inferior to those on FewAsp (random). FewAsp (multi) presents a more challenging scenario than FewAsp (random) because it includes masses of multi-aspect sentences. In most cases, the proposed method still obtains remarkable improvement over other methods on FewAsp (multi), esp. fewer sample scenarios. Compared with the best baseline, we achieve an average of 1.84% AUC improvement on 2-shot and 3-shot settings. Generally, more classes contain more noise. For 15 and 20 classes, the results surpass strong baselines by 2.42% at most, with an average of 1.79% AUC. Besides, we improve the performance better on the "20-way 3-shot" than on the "5-way 3-shot". The fact indicates that the proposed method can alleviate the noises to address the noisy MACD task.

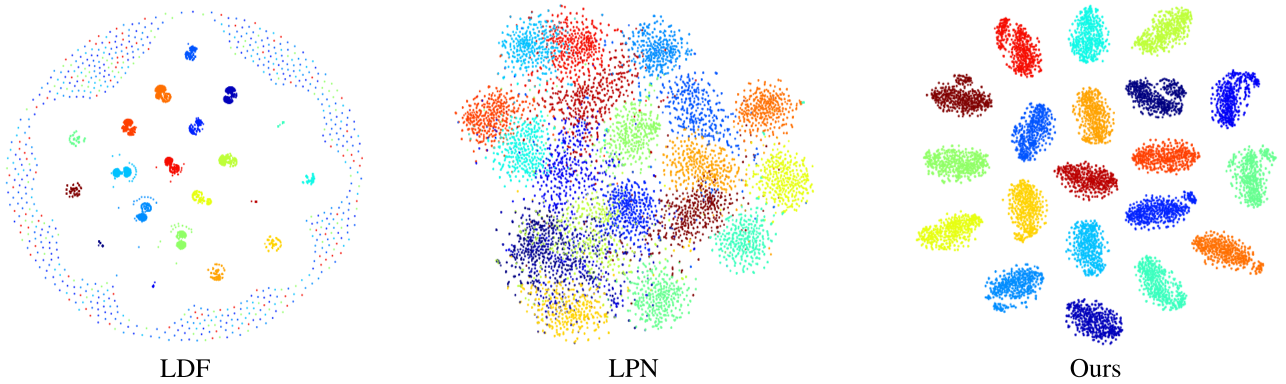
Model	AUC	△ AUC	F1	△ F1
Full model	**95.14**		**77.44**	
w/o *FSO*	93.10	-2.04	71.22	-6.22
w/o M_{sub}	93.80	-1.34	76.02	-1.42
w/o M_{uni}	94.11	-1.03	77.18	-0.26
w/o M_{sub} & M_{uni}	94.74	-0.40	76.89	-0.55

Table 7: Comparison of AUC and Macro-F1 for ablation study on the 5-way 2-shot scenario from FewAsp (multi).

(3) Under the Macro-F metric, we obtain at most 11.01% and an average of 8.59% Macro-F improvements when there are two or three samples. Therefore, the proposed method is superior to the strong baselines LPN [Zhao *et al.*, 2022] and LDF [Liu *et al.*, 2022] for multi-label prediction. To achieve multi-label results, LPN trains an adaptive multi-label module to determine the threshold, and LDF utilizes empirical knowledge to make threshold selections. However, these methods are unfaithful in fewer sample scenarios. The proposed method can address the noisy MACD task in fewer sample scenarios, avoiding complex threshold training or selection. In conclusion, all experimental results verify the effectiveness of our method.

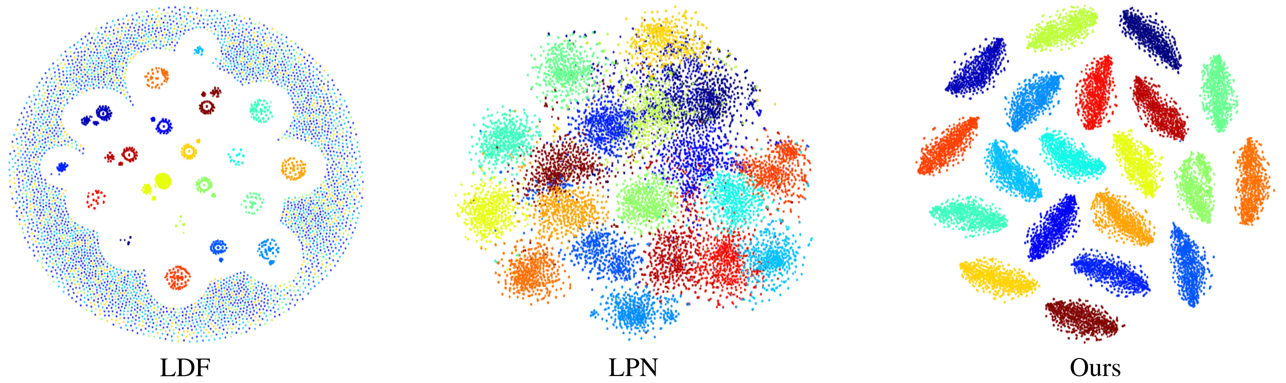
4.3 Ablation Study

We conduct an ablation study on the 5-way 2-shot scenario from FewAsp (multi). A set of ablation experiments are implemented to examine the structure design of the proposed method. The detailed results are depicted in Table 7. First, the performance drops by 2.04% AUC and 6.22% Macro-F1 when we remove the FSO. The result indicates that the FSO positively solves the noisy MACD task by learning the interactions among the M_{int}, M_{sub}, and M_{uni} networks. Then, we further implement the ablation study on the FSO. Without M_{sub} or M_{uni} network, the performance of the method is reduced. Besides, the performance is worse when we remove



LDF LPN Ours

Figure 2: Visual comparisons of prototype embeddings for the 5-way 2-shot scenario from FewAsp (multi) in feature space.



LDF LPN Ours

Figure 3: Visual comparisons of prototype embeddings for the 10-way 2-shot scenario from FewAsp (multi) in feature space.

the M_{sub} and M_{uni} networks at the same time. These negative results denote that the interactions among these three networks are of the essence to solving the MACD task. Therefore, any absence of the FSO can decrease our performance. In conclusion, the design of our method is reasonable and achieves the best performance.

4.4 Visualizations

To further analyze the performance, we visualize the embedding representations of prototypes in the feature space compared with dominant LDF and LPN. Specifically, we sample 3000 episodes in the testing set of FewAsp (multi) and visualize the prototype per class by using the visual tool T-SNE [Laurens and Hinton, 2008]. The prototypes are generated by intra-class sentences and are closely related to the target aspects. Visualizing the embedding representations of prototypes, we leverage colored spots to observe the prototype distribution. To fair comparison, we set the seed is 15 and perplexity is 30 for all methods. The results are presented in Figure 2 (i.e., 5-way 2-shot) and Figure 3 (i.e., 10-way 2-shot). For 20 classes from the testing set, we clearly separate the prototype embeddings in the feature space. The LDF has many spots scattered around clusters, whereas the LPN includes many fuzzy clusters. Compared with LDF and LPN, the results denote that our method can eliminate the negative effect caused by noises to extract representative prototypes.

5 Conclusion

We propose a simple yet effective FSO method to solve the noisy MACD task by distinguishing the semantic contents of samples. Learning the interactions among the M_{int}, M_{sub}, and M_{uni} networks, the FSO imitates arithmetic operations on samples to distinguish relevant and irrelevant aspects, which aims to automatically analyze the semantic contents of samples. Eliminating the negative effect caused by noises, the FSO extracts discriminative prototypes and customizes the corresponding dedicated query vector for each class. Meanwhile, we design a multi-label architecture integrated with score-wise loss and multi-label loss to optimize the FSO for multi-label prediction, avoiding complex threshold training or selection. Extensive experiments on benchmark datasets show that the proposed method obtains convincing improvements on the noisy MACD task, esp. fewer sample scenarios.

Acknowledgements

This work is supported by the National Key Research and Development Program of China (No. 2021ZD0111202), the Science and Technology Project of State Grid Corporation of China (No. 5700-202341285A-1-1-ZN), the National Natural Science Foundation of China (No.62176005), PKU-Haier “Smart Home” Qingdao Joint Innovation Lab.

References

[Alfassy *et al.*, 2019] Amit Alfassy, Leonid Karlinsky, Amit Aides, Joseph Shtok, Sivan Harary, Rogerio Feris, Raja Giryes, and Alex M Bronstein. Laso: Label-set operations networks for multi-label few-shot learning. In *Proc. of CVPR*, pages 6548–6557, 2019.

[Allen *et al.*, 2019] Kelsey Allen, Evan Shelhamer, Hanul Shin, and Joshua Tenenbaum. Infinite mixture prototypes for few-shot learning. In *ICML*, pages 232–241. PMLR, 2019.

[Assran *et al.*, 2022] Mahmoud Assran, Mathilde Caron, Ishan Misra, Piotr Bojanowski, Florian Bordes, Pascal Vincent, Armand Joulin, Mike Rabbat, and Nicolas Ballas. Masked siamese networks for label-efficient learning. In *Proc. of ECCV*, pages 456–473. Springer, 2022.

[Bauman *et al.*, 2017] Konstantin Bauman, Bing Liu, and Alexander Tuzhilin. Aspect based recommendations: Recommending items with the most valuable aspects based on user reviews. In *Proc. of ACM SIGKDD*, pages 717–725, 2017.

[Cai *et al.*, 2020] Hongjie Cai, Yaofeng Tu, Xiangsheng Zhou, Jianfei Yu, and Rui Xia. Aspect-category based sentiment analysis with hierarchical graph convolutional network. In *Proc. of ICCL*, pages 833–843, 2020.

[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 *Proc. of NAACL-HLT*, pages 4171–4186, 2019.

[Gao *et al.*, 2019] Tianyu Gao, Xu Han, Zhiyuan Liu, and Maosong Sun. Hybrid attention-based prototypical networks for noisy few-shot relation classification. In *Proc. of AAAI*, volume 33, pages 6407–6414, 2019.

[Hospedales *et al.*, 2021] Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. Meta-learning in neural networks: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(9):5149–5169, 2021.

[Hu *et al.*, 2021] Mengting Hu, Shiwan Zhao, Honglei Guo, Chao Xue, Hang Gao, Tiegang Gao, Renhong Cheng, and Zhong Su. Multi-label few-shot learning for aspect category detection. In *Proc. of ACL*, 2021.

[Laurens and Hinton, 2008] Van Der Maaten Laurens and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(2605):2579–2605, 2008.

[Lee *et al.*, 2019] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *Proc. of CVPR*, pages 10657–10665, 2019.

[Li *et al.*, 2020] Yuncong Li, Cunxiang Yin, Sheng-hua Zhong, and Xu Pan. Multi-instance multi-label learning networks for aspect-category sentiment analysis. In *Proc. of EMNLP*, pages 3550–3560, 2020.

[Liu *et al.*, 2022] Han Liu, Feng Zhang, Xiaotong Zhang, Siyang Zhao, Junjie Sun, Hong Yu, and Xianchao Zhang. Label-enhanced prototypical network with contrastive learning for multi-label few-shot aspect category detection. In *Proc. of ACM SIGKDD*, pages 1079–1087, 2022.

[Lv *et al.*, 2021] Hui Lv, Chen Chen, Zhen Cui, Chunyan Xu, Yong Li, and Jian Yang. Learning normal dynamics in videos with meta prototype network. In *Proc. of CVPR*, pages 15425–15434, 2021.

[Pontiki *et al.*, 2016] Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, and Manandhar. Semeval-2016 task 5: Aspect based sentiment analysis. In *Proc. of SemEval 2016*, pages 19–30, 01 2016.

[Schmitt *et al.*, 2018] Martin Schmitt, Simon Steinheber, Konrad Schreiber, and Benjamin Roth. Joint aspect and polarity classification for aspect-based sentiment analysis with end-to-end neural networks. In *Proc. of EMNLP*, 2018.

[Snell *et al.*, 2017] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, volume 30, 2017.

[Sun *et al.*, 2020] Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. Circle loss: A unified perspective of pair similarity optimization. In *Proc. of CVPR*, pages 6398–6407, 2020.

[Sung *et al.*, 2018] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proc. of CVPR*, pages 1199–1208, 2018.

[Tsendsuren and Hong, 2017] Munkhdalai Tsendsuren and Yu Hong. Meta networks. In *Proc. of ICML*, pages 2554–2563, 2017.

[Tulkens *et al.*, 2020] Tulkens, Cranenburgh, and andAndreas. Embarrassingly simple unsupervised aspect extraction. In *Proc. of ACL*, 2020.

[Wang *et al.*, 2021] Runzhong Wang, Junchi Yan, and Xiaokang Yang. Neural graph matching network: Learning lawler's quadratic assignment problem with extension to hypergraph and multiple-graph matching. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021.

[Yan *et al.*, 2022] Kun Yan, Chenbin Zhang, Jun Hou, Ping Wang, Zied Bouraoui, Shoaib Jameel, and Steven Schockaert. Inferring prototypes for multi-label few-shot image classification with word vector guided attention. In *Proc. of AAAI*, 2022.

[Yang *et al.*, 2020] Zhuo Yang, Yufei Han, Guoxian Yu, Qiang Yang, and Xiangliang Zhang. Prototypical networks for multi-label learning. In *Proc. of AAAI*, 2020.

[Yu *et al.*, 2021] Weiping Yu, Taojiannan Yang, and Chen Chen. Towards resolving the challenge of long-tail distribution in uav images for object detection. In *Proc. of WACV*, pages 3258–3267, January 2021.

[Zhao *et al.*, 2022] Fei Zhao, Yuchen Shen, Zhen Wu, and Xinyu Dai. Label-driven denoising framework for multi-label few-shot aspect category detection. *Findings of EMNLP*, 2022.