Multi-label Few-shot Learning with Semantic Inference

Zhen Wang,[1] **Yiqun Duan,**[1] **Liu Liu,**[1] **Dacheng Tao**[1]
[1]UBTECH Sydney AI Centre, School of Computer Science, Faculty of Engineering
The University of Sydney, Darlington, NSW 2008, Australia
zwan4121@uni.sydney.edu.au

Abstract

Few-shot learning can adapt the classification model to new labels with only a few labeled examples. Previous studies mainly focus on the scenario of a single category label per example but have not solved the more challenging multi-label scenario with exponential-sized output space and low-data effectively. In this paper, we propose a semantic-aware meta-learning model for multi-label few-shot learning. Our approach can learn and infer the semantic correlation between unseen labels and historical labels to quickly adapt multi-label tasks from only a few examples. Specifically, features can be mapped into the semantic embedding space via label word vectors to explore and exploit the label correlation, and thus cope with the challenge on the overwhelming size of the output space. Then a novel semantic inference mechanism is designed for leveraging prior knowledge learned from historical labels, which will produce good generalization performance on new labels to alleviate the low-data problem. Finally, extensive empirical results show that the proposed method significantly outperforms the existing state-of-the-art methods on the multi-label few-shot learning tasks.

Introduction

Few-shot learning is proposed to facilitate deep learning systems to learn new concepts with very limited labeled data, which is experiencing rapid development and advancements. Most of the latest few-shot learning works (Finn and et al. 2017; Rusu and et al. 2019) are based on meta-learning (*learning-to-learn*) paradigm. Meta-learning is a task-level learning framework that aims to accumulate knowledge from learning a large number of tasks and generalize the knowledge to learn new tasks effectively.

Nevertheless, most of the previous works only focus on the scenario where each example is associated with exclusive single label, but ignore the more actual and challenging scenario, in which each example can be simultaneously associated with multiple labels. The key challenge of multi-label few-shot learning (ML-FSL) is the huge output space, where the number of possible label sets exponentially grows with the increasing number of category labels. The huge output space simultaneously denotes a much sparser learning target and will bring difficulties in model learning for ML-FSL. Moreover, considering the fact that few-shot learning requires the model to be optimized on a relatively small dataset (Antoniou, Edwards, and Storkey 2019), a severe over-fitting problem will arise, which would also be aggravated by the huge output space as well.

In this paper, we propose a gradient-based meta-learning framework, Semantic Inference Network (SIN), to solve the ML-FSL problem. Different from previous meta-learning approaches, we utilize word embedding vectors instead of one-hot vectors as our prediction output to map features into semantic space, which gives us a tighter learning object. More specifically, our approach trains a semantic-aware feature extractor that maps features into the semantic embedding space via label word vectors learned from unsupervised text corpora. That is, the semantic correlation across labels can be preserved by the base model, which helps to structure and regularize the overwhelming output space of ML-FSL. Furthermore, we propose a meta-learning framework with a semantic inference mechanism that can extract semantic features and exploit the correlation between novel labels and historical labels as prior knowledge to classify multiple labels only using a few examples effectively. The semantic inference mechanism has two functions: training better initialization parameters of model by leveraging the knowledge learned from historical tasks; inferring the classification of novel labels according to the semantic correlation with historical labels. Experimental results suggest that with the help of semantic inference, our model achieves state-of-the-art performance on ML-FSL, and ablation studies validate each module's effectiveness[1].

Methodology

Problem Definition

ML-FSL aims to learn a model that can be well adapted to novel multi-label tasks using only a few annotated examples. To be specific, we are given a sufficient labeled training set associated with a base label set L_{base}. Meanwhile, we also have a testing set with a disjoint set of novel labels L_{novel}, where each label is associated with only a few labeled examples. The **goal** of ML-FSL is to obtain a good multi-label classifier for the novel labels only using a few labeled examples. The huge output space of multi-label prediction and

[1]code and models are available on https://github.com/DESIRE-Net/SIN

very few labeled examples limit to train a supervised classification model. To this end, we propose a meta-learning framework, Semantic Inference Network, to explore and exploit the semantic label correlation and meta-knowledge for model novel labels.

Semantic Inference Network

Semantic Inference Network (SIN) consists of two main components: a semantic-aware feature extractor and meta-learner with semantic inference mechanism.

First, we train a semantic-aware feature extractor $\mathcal{F}$ with a output matrix W_{base} on training set to embed features into semantic space, where W_{base} is assigned and fixed to base-label word vectors learned from unsupervised text corpora. Through training with cross-entropy loss on training set, the features produced by feature extractor (i.e., $\boldsymbol{z}{=}\mathcal{F}(\boldsymbol{x})$) would be embedded into a semantic space that naturally makes the output space of ML-FSL tighter, thus alleviate the overwhelming output space problem.

Second, a semantic inference mechanism is designed for leveraging the prior knowledge learned from base labels to adapt new multi-label tasks. Since we have obtained features embedded into the semantic space, the output feature $\boldsymbol{z}$ would naturally be closer to those with similar semantic meanings. In that case, we propose to infer our predictions by proximity between $\boldsymbol{z}$ and novel label word vectors W_{novel}, and use the semantic correlation between base-label word vectors and novel-label word vectors. The semantic inference mechanism $\mathcal{I}$ takes the form:

$$\mathcal{I}(\boldsymbol{z}){=}\left(\gamma_1\frac{\mathcal{W}_Z(\boldsymbol{z})}{\|\mathcal{W}_Z(\boldsymbol{z})\|_2}{+}\gamma_2\frac{\boldsymbol{z}W_{base}}{\|\boldsymbol{z}W_{base}\|_2}W_I\right)W_{novel}\ , \quad (1)$$

where $\mathcal{W}_Z(\boldsymbol{z})$ denotes a nonlinear transformation for $\boldsymbol{z}$, and $\|\cdot\|_2$ denotes l_2 normalization which can eliminate the influence of the absolute magnitudes of semantic features and improve the robustness; W_I is a learnable matrix trained on different tasks to learn deep correlation between novel and base labels; γ_1 and γ_2 are the learnable module factors.

We adopt a meta-learning training process. In each iteration, a few examples are sampled for training to compute and update the meta-learner parameters that will achieve the maximal possible performance on the new task.

Experiments

Datasets and Evaluation

We conduct experiments on a widely used multi-label dataset, Delicious[2], which has 983 labels and 16105 examples. We split the dataset into three parts where labels are mutually disjoint: training set including 600 labels, validation set including 175 labels, testing set including 200 labels. For an N-way K-shot testing setting, each testing task is sampled with N labels, and each label includes K labeled examples and 15 testing examples. The testing results was evaluated on AUC.

[2]http://mulan.sourceforge.net/

Result Analysis and Ablation Study

Table 1 illustrates the average performance of our model in comparison with meta-learning baselines: MAML (Finn and et al. 2017), Reptile (Nichol, Achiam, and Schulman 2018), ATAML (Jiang and et al. 2018), MAML++ (Antoniou, Edwards, and Storkey 2019) and LEO (Rusu and et al. 2019).

Method	5-way AUC		10-way AUC	
	1-shot	5-shot	1-shot	5-shot
pre-trained	67.5%	75.3%	73.0%	81.7%
MAML	75.9%	81.9%	79.0%	82.3%
Reptile	76.1%	81.6%	79.2%	84.2%
ATAML	83.1%	82.9%	83.0%	85.1%
MAML++	81.3%	84.0%	80.6%	84.4%
LEO	85.8%	87.2%	84.8%	86.8%
SIN$\backslash l_2$-norm	84.0%	88.5%	83.8%	86.1%
SIN$\backslash\mathcal{F}$	80.1%	85.8%	78.6%	83.5%
SIN$\backslash\mathcal{I}$	78.2%	84.3%	77.2%	83.8%
SIN(ours)	**87.2%**	**90.5%**	**87.9%**	**88.7%**

Table 1: Comparing multi-label few-shot classification performance on Delicious. Ablation study: $\backslash l_2$-norm denotes l_2 normalization is removed from our model; $\backslash\mathcal{F}$ denotes feature extractor is removed; $\backslash\mathcal{I}$ denotes semantic inference mechanism is removed.

The results consistently shows that the proposed model SIN outperforms the baselines on AUC metric for both 1-shot and 5-shot, 5-way and 10-way. Our approach can extract semantic features and exploit the correlation between novel labels and base labels as prior knowledge, therefore, it can achieve better results in dealing with the problem of multi-label few-shot learning. Ablation studies as shown in Table 1 validate effectiveness of each proposed module: l_2 normalization, semantic-aware feature extractor, and semantic inference mechanism. More detailed experimental results are given in the Supplemental Material.

Conclusion

In this paper, we propose a semantic inference network for multi-label few-shot learning, which can quickly adapt multi-label classification tasks using only a few labeled examples. Experiments illustrate that our approach could achieve best results amongst other meta-learning models by introducing our proposed semantic inference mechanism.

References

Antoniou, A.; Edwards, H.; and Storkey, A. 2019. How to train your MAML. In *ICLR*.

Finn, C.; and et al. 2017. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In *ICML*.

Jiang, X.; and et al. 2018. On the Importance of Attention in Meta-Learning for Few-Shot Text Classification. *ArXiv* .

Nichol, A.; Achiam, J.; and Schulman, J. 2018. On First-Order Meta-Learning Algorithms. *ArXiv* abs/1803.02999.

Rusu, A. A.; and et al. 2019. Meta-Learning with Latent Embedding Optimization. In *ICLR*.

Supplementary Material:
Multi-label Few-shot Learning with Semantic Inference

In this supplementary material, we provide detailed related work, problem definition, model architecture, experimental settings and results. Appendix A provides related work. Appendix B gives the mathematical definition of ML-FSL, illustrates a architecture for the proposed model, and describes the process of training and testing. Appendix C presents implementation details, definition of evaluation metrics, and baselines. Appendix D presents additional empirical evaluation for multi-label few-shot learning and further analyses.

A Related Work

The goal of few-shot learning is that adapting the model to predict new labels with only a few labeled examples effectively. Early works build a Bayesian model with prior knowledge learned from previous labels that can be transferred to new labels [1, 2]. Recently, meta-learning (*learning-to-learn*) [3, 4, 5, 6] framework has been applied to few-shot learning problem and exerted significant impacts by training a meta-learner with few-shot tasks sampled from training data set. Meta-learning methods for few-shot learning problem can be mainly categorized into two groups: metric-based methods and gradient-based methods.

Metric-based methods [7, 8, 9, 10, 11] train a model to embed examples into a metric space where examples with same label are gathered closely, and examples with different labels are spread far away. How to define and learn the similarity among data examples in the metric space is the key for this strategy. For instance, Matching Network [7] can distinguish different labels by a weighted nearest neighbor classifier. Prototypical Network [8] can firstly train an average value of the features belonged the same label in the metric space as the prototype, and then perform nearest neighbor classification. The works [10, 11] further generalize prototypical network with a task adaptive metric [10] and a representation transfer [11]. Sung et al. [9] propose to use relation networks to learn the distance metric for the images within each task. All these metric-based methods can allocate an example to a single label through the maximum similarity (the nearest neighbor), and they can not treat the example associated with multi-label, e.g., an image including dog, cat, and trees, etc., can be identified simultaneously.

Gradient-based methods [5, 6, 12, 13, 14, 15] aim at training parameters of models that can be well adapted to novel tasks with only a few optimization updates. Finn et al. [5] propose a model-agnostic meta-learning (MAML) framework. In order to improve the efficiency and performance of MAML on few-shot learning, many follow-up works built on top of MAML [16, 17]. Reptile [12] simply replace the second-order gradient information with the first-order gradient computation of MAML. MAML++ [14] further improve the generalized performance and stabilize the system. ATAML [13] is designed to encourage task-agnostic representation learning with attention mechanisms. LEO [15] employs a low-dimensional latent space to learn the model parameters. Current approaches mentioned above only mainly focus on multi-class (single-label) few-shot learning, but ignores the multi-label problem. Although gradient-based approaches can be easily adapted to use in multi-label problem by converting 1-hot output to n-hot output, the challenges brought by the exponential-sized output space and more serious problem such as low-data will severely restrict the performance of meta-learner.

Multi-label few-shot learning (ML-FSL) aims to adapt the model to new multi-label classification tasks with only a few labeled examples. Up to now, only a few works have been attempted to address this challenging problem. ZAG-CNN [18] aims at solving *long-tail* dataset with infrequent labels (few-shot data), but has not addressed a complete ML-FSL problem. LaSO [19] points out the difficulties in ML-FSL and then proposes a data-augment technique to generalize different label sets. However, LaSO only verifies the validity of data-augment and can not be applied to solve ML-FSL. In order to fill in the above mentioned research gaps, this paper propose a gradient-based meta-learning model, Semantic Inference Network (SIN), to solve ML-FSL effectively.

B Problem definition and Model Architecture

In this section, after giving the mathematical definition of the proposed multi-label few-shot learning (ML-FSL) problem, we provide a architecture figure for the Semantic Inference Network (SIN).

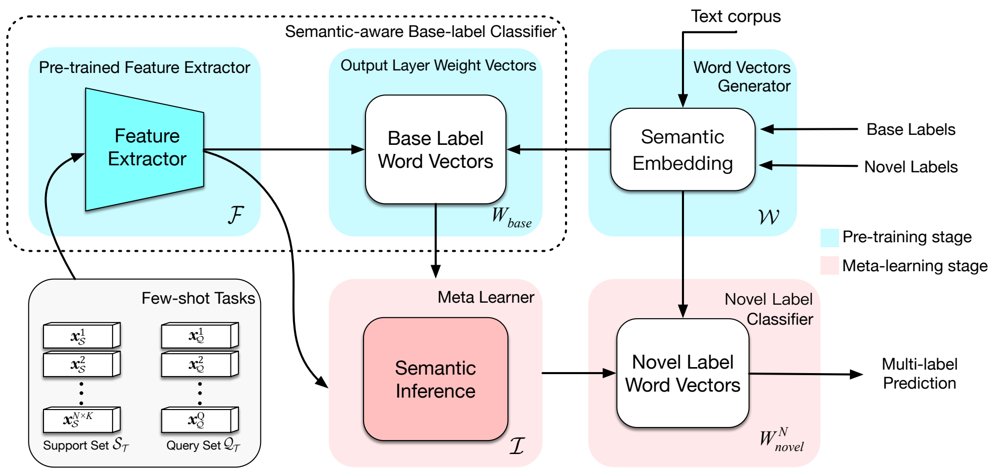

Figure 1: Semantic Inference Network for multi-label few-shot learning. In the pre-training stage, we learn a semantic-aware base-label classifier with a feature extractor to embed features into the semantic space. In the meta-learning stage, we train a meta-learner with a semantic inference mechanism on different few-shot tasks to incorporate semantic knowledge. In the testing stage, we evaluate the model on novel tasks.

B.1 Multi-label Few-shot Learning

ML-FSL aims to learn a model that can be well adapted to novel multi-label tasks using only a few annotated examples. We formulate ML-FSL as a meta-learning problem and adopt the episode training mechanism by following the convention [7, 5, 8, 13, 14, 15]. Generally, in an N-way K-shot episode setting, each task $\mathcal{T}$ is formed by first sampling N labels from $\mathcal{D}_{meta}$ and then sampling two sets of examples associated with these labels: 1) the *support* set $\mathcal{S}_\mathcal{T} = \{(\boldsymbol{x}^i_\mathcal{S}, \boldsymbol{y}^i_\mathcal{S})\}^{N\times K}_{i=1}$ containing K examples sampling from each of the N labels, and 2) the *query* set $\mathcal{Q}_\mathcal{T} = \{(\boldsymbol{x}^i_\mathcal{Q}, \boldsymbol{y}^i_\mathcal{Q})\}^{Q}_{i=1}$ containing a fraction of the rest examples from the same N labels. Here, $\boldsymbol{y}^i \in \mathcal{Y} \subseteq \{0, 1\}^{N\times 1}$ denotes the multi-label vector, $\boldsymbol{x}^i \in \mathcal{X} \subseteq \mathbb{R}^{d_s \times 1}$ denotes the corresponding example point, and d_s is the dimension of the example. If the j-th label is assigned to example $\boldsymbol{x}^i$, we have $y^{i,j} = 1$, otherwise $y^{i,j} = 0$. The proposed meta-learning problem is to build a mapping f_θ (known as meta-learner) from the support set and the query set examples to the query set labels.

The resulting meta-learner objective is to minimize, over all tasks, the expected loss of the prediction on examples in query set, given the support set:

$$\theta = \arg\min_\theta \mathop{\mathbb{E}}_{\mathcal{T}\sim\mathcal{D}_{meta}} [\![\mathcal{L}(\mathcal{T};\theta)]\!], \tag{1}$$

where

$$\mathcal{L}(\mathcal{T};\theta) = \mathop{\mathbb{E}}_{\mathcal{S}_\mathcal{T},\mathcal{Q}_\mathcal{T}} \left[\!\!\left[\sum_{q=1}^{|\mathcal{Q}_\mathcal{T}|} \ell\left(\boldsymbol{y}^q_\mathcal{Q}, f_\theta(\boldsymbol{x}^q_\mathcal{Q}, \mathcal{S}_\mathcal{T})\right) \right]\!\!\right], \tag{2}$$

$(\boldsymbol{x}^q_\mathcal{Q}, \boldsymbol{y}^q_\mathcal{Q}) \in \mathcal{Q}_\mathcal{T}$ and $\mathcal{S}_\mathcal{T}$ are, respectively, the query and support set sampled from $\mathcal{D}_{meta}$, θ are the parameters of the model, and $\ell(\cdot)$ denotes the loss function.

In meta-learning framework, we typically have different meta-sets $\mathcal{D}_{meta-train}$, $\mathcal{D}_{meta-val}$, and $\mathcal{D}_{meta-test}$, for meta-training, meta-validation, and meta-testing, respectively. The label sets are disjoint among these meta-sets, where the labels belonged to $\mathcal{D}_{meta-train}$ are called **base labels** and the labels belonged to $\mathcal{D}_{meta-test}$ are called **novel labels**. The meta-learner is trained on the tasks sampled from $\mathcal{D}_{meta-train}$ by minimizing Equation (2). We can select the hyper-parameters of the meta-learner on $\mathcal{D}_{meta-val}$ and finally evaluate generalization performance on $\mathcal{D}_{meta-test}$.

B.2 Semantic Inference Network Architecture

Figure 1 shows the structure of SIN. As a preliminary preparation, we train a word-embedding model $\mathcal{W}$ on unsupervised text corpora, e.g., Google News. We employ $\mathcal{W}$ to produce label word embeddings

Algorithm 1 Semantic Inference Network: Meta Training

Require: Meta-train set $D_{meta-train}$
Require: Learning rates α, β
Train feature extractor $\mathcal{F}$ on $\mathcal{D}_{meta-train}$
Initialize θ and $\boldsymbol{\gamma} = \{\gamma_1, \gamma_2\}$ ▷ Initialize all parameters
while not done **do**
  Sample batch of tasks $\mathcal{T}_i \sim \mathcal{D}_{meta-train}$ ▷ Sample tasks for meta-training
  Let $(\mathcal{S}_{\mathcal{T}_i}, \mathcal{Q}_{\mathcal{T}_i}) = \mathcal{T}_i$ ▷ Get support set and query set
  for all $\mathcal{T}_i$ **do**
    $\theta_i' = \theta - \alpha \nabla_\theta \mathcal{L}(\mathcal{S}_{\mathcal{T}_i}; \{\theta, \boldsymbol{\gamma}\})$ ▷ Compute temporary parameters
  end for
  Update $\theta \leftarrow \theta - \beta \nabla_\theta \sum_{\mathcal{T}_i} \mathcal{L}(\mathcal{Q}_{\mathcal{T}_i}; \{\theta_i', \boldsymbol{\gamma}\})$ ▷ Update network parameters
  Update $\boldsymbol{\gamma} \leftarrow \boldsymbol{\gamma} - \beta \nabla_{\boldsymbol{\gamma}} \sum_{\mathcal{T}_i} \mathcal{L}(\mathcal{Q}_{\mathcal{T}_i}; \{\theta_i', \boldsymbol{\gamma}\})$ ▷ Update factor parameters
end while

of all labels in $\mathcal{D}_{meta-train}$ and $\mathcal{D}_{meta-test}$. Our model consists of two main components. First, we train a semantic-aware base-label classifier on $\mathcal{D}_{meta-train}$, where the output matrix of the classifier is replaced by word embedding vectors of all base labels. Thus, as a part of the classifier, the feature extractor could embed features into semantic space, which simultaneously facilitate the model to learn the semantic correlation between labels. Second, a gradient-based semantic inference mechanism is designed for leveraging the prior knowledge learned from base labels to adapt new multi-label tasks on $\mathcal{D}_{meat-test}$.

B.3 Meta Training and Meta Testing

In this section, we provide meta training and testing procedures of our proposed framework.

Meta Training. The training algorithm is described in Algorithm 1. We use θ to denote the parameters of the meta-learner including those of learnable matrices and networks, and use $\boldsymbol{\gamma} = \{\gamma_1, \gamma_2\}$ to denote the learnable factors. It is noted that the parameters of trained feature extractor are frozen. In meta training, each N-way K-shot task $\mathcal{T}_i = \{\mathcal{S}_{\mathcal{T}_i}, \mathcal{Q}_{\mathcal{T}_i}\}$ is sampled from $\mathcal{D}_{meta-train}$.

The goal of meta-learner is to obtain good parameters for θ and $\boldsymbol{\gamma}$ that can be adapted to different tasks using a few optimization steps. To achieve this goal, gradient-based methods minimize the loss of the prediction on query set $\mathcal{Q}_{\mathcal{T}_i}$, given the support set $\mathcal{S}_{\mathcal{T}_i}$. First, we compute a *inner* loss on support set $\mathcal{S}_{\mathcal{T}_i}$ to update the parameters θ and get a temporary parameters θ_i'.

$$\theta_i' = \theta - \alpha \nabla_\theta \mathcal{L}(\mathcal{S}_{\mathcal{T}_i}; \{\theta, \boldsymbol{\gamma}\}) \tag{3}$$

Then, an *outer* loss is computed using θ_i' and $\boldsymbol{\gamma}$ on query set $\mathcal{Q}_{\mathcal{T}_i}$ to update meta-learner parameters θ_i and $\boldsymbol{\gamma}$. Different from the traditional gradient-based method [5], part of parameters, factors $\boldsymbol{\gamma}$, in our model can be reused across all tasks without being adapted on the support set. The novel training paradigm provides regularization of meta-learning that further improves generalization.

An interesting setting in our training is that we pick "fake" novel labels from base labels $L_{base} \in \mathcal{D}_{meta-train}$ to learn how to treat actual novel labels $L_{novel} \in \mathcal{D}_{meta-test}$. Specifically, for one training task, we sample N labels from L_{base} as "fake" novel labels, and then take these N labels out of L_{base}. The remaining L_{base} serves as base labels. Accordingly, we need to dynamically eliminate corresponding N of m vectors in the inference matrix W_I during training. Everything is trained end-to-end.

Meta Testing. Meta testing aims to test the performance of the trained SIN for fast adaptation to novel tasks sampled from $\mathcal{D}_{meta-test}$. Given $\mathcal{T}_{novel} = \{\mathcal{S}, \mathcal{Q}\}$, we fine-tune the SIN on $\mathcal{S}$, and test on $\mathcal{Q}$.

C Experimental Settings

C.1 Implementation Details

Pytorch[1] is used to implement the proposed algorithm and to conduct all the experiments. All the computations are performed on a 64-Bit Linux workstation with 10-core Intel Core CPU i7-6850K 3.60GHz processor, 256 GB memory, and 4 Nvidia GTX 1080 Ti GPUs. For meta-training stage, we use Adam optimizer with a fixed learning rate 0.001, weight decay 10^{-6}, $\beta_1 = 0.9$ and $\beta_2 = 0.99$.

We conduct experiments on a widely used multi-label dataset, Delicious[2], which has 983 labels and 16105 examples. We removed 8 punctuation labels without semantics and split the dataset into three parts where labels are mutually disjoint: $\mathcal{D}_{meta-train}$ including 600 labels, $\mathcal{D}_{meta-val}$ including 175 labels, $\mathcal{D}_{meta-test}$ including 200 labels. For an N-way K-shot setting, each task is sampled with N labels, and each label includes K support examples and 15 query examples. Note that due to the label distribution of multi-label data [20], several labels may correspond to more examples than other labels, however, the comparison on different methods are fair because of the same setting. We use GloVe [21] to generate the word vectors for the category labels as the semantic embeddings. The GloVe model is trained with large unsupervised text corpora.

We train models 100 epochs, where each epoch contains 1,000 tasks randomly sampled from $\mathcal{D}_{meta-train}$. For the meta-testing stage, we test models on 1,000 novel tasks randomly sampled from $\mathcal{D}_{meta-test}$ to get average results. The semantic embedding model, GloVe [21], generates 300-dimension word vectors for the category labels[3]. For the architecture of SIN, the feature extractor $\mathcal{F}$ has 3 layers of fully connected layers, and the meta-learner $\mathcal{I}$ consists of different networks with several connected layers. The dropout rate in the networks is set in the range $[0.1, 0.5]$. All hyperparameters are cross-validated in validation set $\mathcal{D}_{meta-val}$ and fixed afterward in all experiments. We also provide the source code for reference[4].

C.2 Evaluation Metrics and Settings

Given a query set $\mathcal{Q}_{\mathcal{T}}$ sampled from meta-test dataset $\mathcal{D}_{meta-test}$ denoted by $\mathcal{Q}_{\mathcal{T}} = \{(\boldsymbol{x}_1, \boldsymbol{y}_1), ..., (\boldsymbol{x}_q, \boldsymbol{y}_q)\}$, where $\boldsymbol{x}_i \in \mathcal{X} \subseteq \mathbb{R}^{d_s \times 1}$ is a real vector representing an input feature (instance) and $\boldsymbol{y}_i \in \mathcal{Y} \subseteq \{0,1\}^{N \times 1}$ is the corresponding output label vector ($i \in [n]$, defined as $i \in \{1, ..., n\}$). Moreover, $y_i^j = 1$ if the j-th label is assigned to the instance $\boldsymbol{x}_i$ and $y_i^j = 0$ otherwise. For notational simplicity, we use $Y_{i\cdot}^{+}$ to denote the index set of associated (non-associated) labels of $\boldsymbol{y}_i$. Formally, $Y_{i\cdot}^{+} = \{j | y_i^j = 1\}$ and $Y_{i\cdot}^{-} = \{j | y_i^j = 0\}$. With respect to j-th column of label matrix, $Y_{\cdot j}^{+} = \{i | y_i^j = 1\}$ denotes the index set of associated instances of the j-th label and $Y_{\cdot j}^{-} = \{i | y_i^j = 0\}$ denotes the set of non-associated instances similarly. We use $|\cdot|$ to represent the cardinality of a set.

Table 1: Definitions of multi-label performance measures.

Measure	Formulation	Note
macro-F1	$\textit{macro-F1}(H) = \frac{1}{m} \sum_{j=1}^{m} \frac{2 \sum_{i=1}^{n} y_{ij} h_{ij}}{\sum_{i=1}^{n} y_{ij} + \sum_{i=1}^{n} h_{ij}}$	F-measure averaging on each label.
micro-F1	$\textit{micro-F1}(H) = \frac{2 \sum_{j=1}^{m} \sum_{i=1}^{n} y_{ij} h_{ij}}{\sum_{j=1}^{m} \sum_{i=1}^{n} y_{ij} + \sum_{j=1}^{m} \sum_{i=1}^{n} h_{ij}}$	F-measure averaging on the prediction matrix.
AUC	$\textit{micro-AUC}(F) = \frac{\lvert\mathcal{S}_{\text{micro}}\rvert}{(\sum_{i=1}^{n} \lvert Y_{i\cdot}^{+}\rvert) \cdot (\sum_{i=1}^{n} \lvert Y_{i\cdot}^{-}\rvert)}$ $\mathcal{S}_{\text{micro}} = \{(a, b, i, j) \mid (a, b) \in Y_{\cdot i}^{+} \times Y_{\cdot j}^{-},\ f_i(\boldsymbol{x}_a) \geq f_j(\boldsymbol{x}_b)\}$	AUC averaging on prediction matrix. $\mathcal{S}_{\text{micro}}$ is the set of correct quadruples.

[1] `https://pytorch.org/`

[2] `http://mulan.sourceforge.net/`

[3] `https://nlp.stanford.edu/projects/glove/`

[4] `https://github.com/DESIRE-Net/SIN`

Table 1 summarizes three popular multi-label evaluation metrics used in this paper, which can be divided into bipartition-based metrics, i.e., macro-F1 and micro-F1, and a ranking-based metric, i.e., AUC [20, 22]. We assume that $H : \mathbb{R}^d \to \{0,1\}^m$ is the multi-label classifier and predicts which labels an instance is associated with. H can be decomposed as $\{h^1, ..., h^m\}$ and $h^j(\boldsymbol{x}_i)$ represents the prediction of y_i^j. The results of H can be evaluated by bipartition-based metrics. $F : \mathbb{R}^d \to \mathbb{R}^m$ is the multi-label predictor, whose predicted value could be regarded as the confidence of association. $F = \{f^1, ..., f^m\}$ and $f^j(\boldsymbol{x}_i)$ denotes the predicted value of y_i^j, which can be evaluated by ranking-based metrics. H can be induced from F by thresholding techniques $t(\cdot)$. For example, $h^j(\boldsymbol{x}_i) = \mathbb{1}\{f^j(\boldsymbol{x}_i) > t(\boldsymbol{x}_i)\}$, where we use $\mathbb{1}\{event\}$ to denote the indicator function for $event$. In the experiment, we simply use 0.5 as the threshold for the output of all models. The evaluation metrics implementation is based on scikit-learn tools[5].

C.3 Baselines

We compare the proposed model with five well-established or state-of-the-art few-shot learning algorithms: **MAML** [5], **Reptile** [12], **ATAML** [13], **MAML++** [14], **LEO** [15]. **MAML** is a groundbreaking gradient-based meta-learning method. **Reptiles** propose a *shortest descent* method to further improve efficiency and performance. **ATAML** introduces attention mechanism into meta-learning to learn task-agnostic representation. **MAML++** is the state-of-the-art meta-learning framework for few-shot learning problem, which employ multi-step loss optimization to improve the generalization performance. **LEO** applies pre-trained representations on a low-dimensional latent space instead of the original high-dimensional parameter space, which achieves current state-of-the-art classification performance. Since metric-based meta-learning methods [7, 8, 9] can only output a single label of the nearest neighbor, it can not be competent for multi-label task with an uncertain number of labels.

D Additional Results and Analyses

D.1 Additional Results

The multi-label few-shot classification performance for SIN and other baselines are show in Table 2. We evaluate 5-way 1-shot, 5-way 5-shot, 10-way 1-shot and 10-way 5-shot learning on Micro-F1 and Macro-F1, respectively. The results show that SIN significantly outperforms other methods and get the state-of-the-art performance on multi-label few-shot learning tasks. Specifically, SIN performs better than MAML with an improvement up about 20%, and better than LEO with an improvement up about 5%. LEO also starts from a pre-trained model to extract features and achieve decent performance, but ignores the correlation between labels. SIN with the ability of semantic inference can explore and exploit the label correlation, and thus facilitate to classify multiple labels using only a few training examples. We can find that SIN has larger improvement in 1-shot task than 5-shot task. This is because SIN incorporates the label semantics to effectively overcome the challenge of insufficient examples.

Table 2: Comparing multi-label few-shot classification performance on Delicious

Method	5-way Micro-F1		10-way Micro-F1		5-way Macro-F1		10-way Macro-F1	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
pre-trained	27.8%	37.3%	17.6%	26.9%	19.1%	29.2%	12.7%	19.4%
MAML	34.1%	48.2%	28.4%	38.5%	27.9%	41.7%	22.9%	33.9%
Reptile	35.7%	51.4%	30.2%	40.4%	28.3%	42.1%	24.0%	35.1%
ATAML	43.6%	52.5%	39.5%	42.1%	36.5%	43.2%	33.8%	36.7%
MAML++	38.3%	53.1%	33.0%	43.3%	31.8%	46.0%	27.1%	38.3%
LEO	51.1%	60.7%	45.1%	51.7%	44.0%	53.5%	38.0%	45.9%
SIN(ours)	**56.5%**	**65.8%**	**49.6%**	**55.1%**	**51.3%**	**60.4%**	**42.2%**	**48.3%**

[5] `https://scikit-learn.org/stable/`

D.2 Additional Analyses

Influence of semantic correlation. Based on the results in Table 2 and Ablation study, it can be shown that label semantic correlation is a key for multi-label few-shot learning. We leverage label semantic correlation in both semantic-aware feature extractor $\mathcal{F}$ and meta-learner with semantic inference mechanism $\mathcal{I}$. Exploiting label correlation can facilitate the multi-label learning process to cope with the challenge of the overwhelming size of output space. For instance, if an image has been annotated with label *whale*, the probability of the image being associated with labels *ocean* and *seaweed* would be high, and the image is unlikely to be labeled as *grassland* and *lion*. Moreover, semantic embedding (learned from large unsupervised text corpora) can serve as prior knowledge and context to supplement the label correlation. The proposed SIN incorporates the label correlation not only from the learning process but also from label semantic embedding, which has achieved great success in multi-label few-shot learning. Ablation study shows that If we remove feature extractor $\mathcal{F}$ or semantic inference $\mathcal{I}$ (denoted as SIN$\backslash\mathcal{F}$ or SIN$\backslash\mathcal{I}$, respectively), SIN will produce a significant performance degradation.

Influence of l_2-norm. l_2-norm is a technique that is often used to provide regularities for deep neural networks. However, in our meta-leaner, l_2-norm plays a more critical role. We employ l_2-norm in three different levels of semantic inference. In the proposed semantic inference, l_2-norm is used to eliminate the influence of the absolute magnitudes of semantic features and improve the robustness. In another, l_2-norm not only offers nonlinear operation for feature transformation but also limits the adverse effects of absolute magnitudes fluctuations of $\boldsymbol{z}W_{base}$. Since the number of base labels is different between meta-training and meta-testing, by using l_2-norm, the absolute value of the feature transformation in meta-training and meta-testing can be kept consistent, which is important for the convergence of the model. Ablation demonstrates that if remove the l_2 normalization (denoted as SIN$\backslash l_2$-norm), the performance of the model will be degraded.

References

[1] Li Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4):594–611, 2006.

[2] Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[3] Jurgen Schmidhuber. Evolutionary principles in self-referential learning. on learning now to learn: The meta-meta-meta...-hook. Diploma thesis, Technische Universitat Munchen, Germany, 1987.

[4] S. Thrun. Lifelong learning algorithms. In *Learning to Learn*, pages 181–209, 1998.

[5] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135, International Convention Centre, Sydney, Australia, 2017. PMLR.

[6] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *In International Conference on Learning Representations (ICLR)*, 2017.

[7] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, koray kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems 29*, pages 3630–3638. 2016.

[8] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems 30*, pages 4077–4087. 2017.

[9] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H.S. Torr, and Timothy M. Hospedales. Learning to compare: Relation network for few-shot learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[10] Chen Xing, Negar Rostamzadeh, Boris Oreshkin, and Pedro O O. Pinheiro. Adaptive cross-modal few-shot learning. In *Advances in Neural Information Processing Systems 32*, pages 4847–4857. 2019.

[11] Yinbo Chen, Xiaolong Wang, Zhuang Liu, Huijuan Xu, and Trevor Darrell. A new meta-baseline for few-shot learning. *ArXiv*, abs/2003.04390, 2020.

[12] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *ArXiv*, abs/1803.02999, 2018.

[13] Xiang Jiang, Mohammad Havaei, Gabriel Chartrand, Hassan Chouaib, Thomas Vincent, Andrew Jesson, Nicolas Chapados, and Stan Matwin. On the importance of attention in meta-learning for few-shot text classification. *ArXiv*, abs/1806.00852, 2018.

[14] Antreas Antoniou, Harrison Edwards, and Amos Storkey. How to train your MAML. In *International Conference on Learning Representations*, 2019.

[15] Andrei A. Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. In *International Conference on Learning Representations*, 2019.

[16] Chelsea Finn, Kelvin Xu, and Sergey Levine. Probabilistic model-agnostic meta-learning. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 9537–9548, 2018.

[17] Jaesik Yoon, Taesup Kim, Ousmane Dia, Sungwoong Kim, Yoshua Bengio, and Sungjin Ahn. Bayesian model-agnostic meta-learning. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 7343–7353, 2018.

[18] Anthony Rios and Ramakanth Kavuluru. Few-shot and zero-shot multi-label learning for structured label spaces. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3132–3142, 2018.

[19] 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 *CVPR*, 2019.

[20] Zhen Wang, Liu Liu, and Dacheng Tao. Deep streaming label learning. In *ICML*, pages 378–387. 2020.

[21] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[22] M. Zhang and Z. Zhou. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1819–1837, 2014.