SEMANTIC INFERENCE NETWORK FOR FEW-SHOT STREAMING LABEL LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Streaming label learning aims to model newly emerged labels for multi-label classification systems, which requires plenty of new label data for training. However, in changing environments, only a small amount of new label data can practically be collected. In this work, we formulate and study few-shot streaming label learning (FSLL), which models emerging new labels with only a few annotated examples by utilizing the knowledge learned from past labels. We propose a meta-learning framework, Semantic Inference Network (SIN), which can learn and infer the semantic correlation between new labels and past labels to adapt FSLL tasks from a few examples effectively. SIN leverages label semantic representation to regularize the output space and acquires label-wise meta-knowledge based on the gradient-based meta-learning. Moreover, SIN incorporates a novel label decision module with a meta-threshold loss to find the optimal confidence thresholds for each new label. Theoretically, we demonstrate that the proposed semantic inference mechanism could constrain the complexity of hypotheses space to reduce the risk of overfitting and achieve better generalizability. Experimentally, extensive empirical results and ablation studies illustrate the superior performance of SIN over the prior state-of-the-art methods on FSLL.

1 INTRODUCTION

In many real-world systems, with more in-depth exploration and understanding of data, the number of associated labels gradually increases (Raghavan & Hafez, 2000; Krempl et al., 2014; Zhou, 2016). For instance, when a photo is posted to Facebook or Twitter, the photo will be continuously tagged with new labels by users who are browsing, and the classification system needs to be updated accurately according to the incoming new labels (Dietterich, 2017). As for previous classification systems, on the one hand, independent learning for only new labels would ignore the correlated information from past labels. On the other hand, integrating past labels with new labels to retrain the whole system each time would be prohibitively computationally expensive. Overall, the problem of learning emerging new labels is referred to as *streaming label learning* (You et al., 2016; Wang et al., 2020b).

Exploring the relationship between new labels and past labels is crucial for modelling newly-arrived labels. The intuition is formulated in the pioneering work of You et al. (2016), which assumes that the new label vectors are a linear combination of past label vectors and that the new classifier can inherit the linear combination to obtain the relationship across labels without retraining the historical classifier. Wang et al. (2020b) proposed a DNN-based framework to learn label dependencies and distill knowledge from the historical model, which models new labels more accurately. In these streaming label learning tasks, the newly labeled data is assumed to be sufficiently available, which could be easily violated in changing environments.

In practical applications, collecting large number of emerging new labels is often prohibitively expensive and time-consuming. In such a situation where only a few examples associated with new labels are available, both training from scratch and fine-tuning on the small dataset are likely to cause severe overfitting, leading to generalization issues in streaming label learning approaches. Although few-shot learning methods (Li Fei-Fei et al., 2006; Snell et al., 2017; Finn et al., 2017; Xing et al., 2019) are designed to accommodate limited labeled data, existing methods mainly focus on single-label examples. It is nontrivial for few-shot learning to adapt to the multiple newly-arrived labels, since the output space (the number of possible label sets) exponentially grows with the increasing

number of category labels. Moreover, the label correlation between past labels and new labels ignored by traditional few-shot learning approaches would be crucial for modeling the emerging new labels.

We formulate the above real-world learning problem as *Few-shot Streaming Label Learning* (FSLL), where new labels associated with a few examples are arrived on the fly, to learn a model for the newly-arrived labels with the help of the knowledge learned from past labels. Our motivation stems from the challenges of modeling the new labels in FSLL: overwhelming output space and low-data problem. Inspired by human infant learning, in which linguistic information can help infants recognize new concepts from very few objects (Jackendoff, 1987; Smith, 2003; Smith & Gasser, 2005), we hypothesize that in the context of FSLL, semantic representation from text can be a powerful source of information to distinguish new labels. In a word embedding space, the semantic representation naturally brings correlation across labels, e.g., the label "cake" is semantically close to the label "food". The semantic correlation across labels could help to structure and regularize the overwhelming output space of FSLL. Moreover, we will leverage the correlation from past labels to new labels and acquire knowledge from historical models to alleviate the low-data problem.

In this paper, we propose a meta-learning framework, Semantic Inference Network (SIN), to solve FSLL problem with only a few examples. Unlike using learnable parameters as output layer weights in the previous neural networks, SIN utilizes the fixed semantic representation of labels (learned from the unsupervised text corpora), which empowers the network to map features into a semantic space. We propose a semantic inference mechanism that can extract semantic features and exploit the correlation between new labels and past labels, to learn more disentangled representations. Furthermore, we propose a label decision module with a novel meta-threshold loss function to find the optimal confidence thresholds for each new label. SIN employs a gradient-based meta-learning paradigm to learn label-wise meta-knowledge while fast adapting semantic inference parameters to distinguish multiple new labels within a few optimization updates.

To verify the effectiveness and the performance of SIN, we conduct both theoretical and experimental analyses. Theoretically, we demonstrate that SIN leverages semantic knowledge and meta knowledge to constrain the complexity of the hypotheses space and reduce the risk of overfitting of FSLL, bringing better generalizability of our proposed model. Experimentally, extensive empirical evaluations show that our model achieves state-of-the-art performance on FSLL, and ablation studies validate the effectiveness of each module. The anonymous code and model are available here[1].

2 Related Work

Streaming Label Learning. As new labels usually emerge from open and dynamic environments (Dietterich, 2017), streaming label learning is proposed to facilitate the learning system with the capability of modeling new labels effectively (You et al., 2016; Wang et al., 2020b). Recent studies are presented to explore and exploit the relationship between past labels and new labels, so that the historical data can be further utilized. SLL (You et al., 2016) learns a mapping from past label vectors to new label vectors and then assume the new classifier can inherit the mapping relationship as a regularization. Constrained by the hypothesis, new classifiers can improve the performance. DSLL (Wang et al., 2020b) is a DNN-based framework to learn deep relationships across labels with a label smoothing technique. Moreover, DSLL has the ability to distill the feature-level knowledge from a past-label classifier to a new-label classifier and outperforms other methods on modeling emerging new labels.

Nevertheless, current streaming label learning approaches assume new labels assigned with a large number of examples, which could be easily violated in changing environments. Consequently, they can not handle new labels only associated with a few examples, which will cause severe overfitting.

Few-shot Learning. Machine learning achieves impressive breakthroughs in data-intensive applications, but it often fails when the data set is small. Few-shot learning (Thrun., 1998; Li Fei-Fei et al., 2006; Chen et al., 2019) is proposed to solve this problem. Current works could be categorized as metric-based and gradient-based approaches. *Metric-based methods* (Vinyals et al., 2016; Snell et al., 2017; Sung et al., 2018; Chen et al., 2020) train a model to embed examples into a metric space where examples with the same label are gathered closely, and examples with different labels are spread far away. For instance, Prototypical Network (Snell et al., 2017) can first train an average value of the features belonging to the same label in the metric space as the prototype and then perform the nearest

[1] https://github.com/ICLR-FSLL/SIN

neighbor classification. AM3 (Xing et al., 2019) introduces semantic representation to adapt the prototype and achieves state-of-the-art performance on few-shot learning. *Gradient-based methods* (Finn et al., 2017; Gidaris & Komodakis, 2018; Antoniou et al., 2019) aim at training parameters of models that can be well adapted to novel tasks with only a few optimization updates. Finn et al. (2017) proposes a model-agnostic meta-learning (MAML) framework, and many follow-up works built on top of MAML (Finn et al., 2018). Reptile (Nichol et al., 2018) simply replaces the second-order gradient information with the first-order gradient computation of MAML. MAML++ (Antoniou et al., 2019) further improve the generalized performance and stabilize the system. ATAML (Jiang et al., 2018) is designed to encourage task-agnostic representation learning with attention mechanisms. LEO (Rusu et al., 2019) applies pre-trained representations on a low-dimensional latent space instead of the original high-dimensional parameter space, which achieves better classification performance.

However, existing few-shot learning methods mainly focus on single-label examples, which is nontrivial to adapt to the multiple newly-arrived labels for solving FSLL problem. For metric-based methods, a testing example can only be allocated to a single label through the maximum similarity (the nearest neighbor), which limits to handle the incoming new labels. For gradient-based methods, although they can be transformed to predict multiple new labels by converting 1-*hot* to n-*hot* output, the challenges brought by the exponential-sized output space would severely restrict the performance on FSLL. Moreover, the label correlation between past labels and new labels is ignored, while label semantic correlation served as the meta-information can be crucial when the available data is scarce.

3 METHODOLOGY

In this section, after giving the mathematical definition of few-shot streaming label learning (FSLL), we give technical details of the proposed Semantic Inference Network (SIN) to handle the newly-emerged labels associated with only a few examples.

3.1 PROBLEM FORMULATION

First, we formulate the problem of FSLL. Assume $\mathcal{D}=\{(\boldsymbol{x}_1,\boldsymbol{y}_1),...,(\boldsymbol{x}_n,\boldsymbol{y}_n)\}$ is an initial training data set, where $\boldsymbol{x}_i\in\mathcal{X}$ is a real vector representing an input feature (example) and $\boldsymbol{y}_i\in\mathcal{Y}_0\subseteq\{0,1\}^{m\times 1}$ is the corresponding output label vector, where m is the number of past labels, $i\in\{1,...,n\}$. Moreover, if the j-th label is associated to the example $\boldsymbol{x}_i$, $y_i^j=1$; otherwise, $y_i^j=0$. By observing $\mathcal{D}$, a proper learning model can be derived for the m past labels. In a streaming fashion, new labels will continuously emerge on a few examples. For simplicity, we denote $\boldsymbol{y}^{new}=[y^{m+1},y^{m+2},...,y^{m+k}]^T\in\mathbb{R}^k$ as the new k-labels vector for an example $\boldsymbol{x}$, where $k\geq 1$. A new label is associated with only N_s examples (usually $N_s<5$). Few-shot streaming label learning aims to derive a learning model for the emerging new labels using only a few labeled examples.

3.2 SEMANTIC INFERENCE NETWORK

We propose Semantic Inference Network (SIN), designed for FSLL, to accommodate emerging new labels using only a few examples. Figure 1 illustrates the overall learning framework of SIN. Different from previous neural networks using learnable parameters as output layer weights, we utilize fixed semantic embedding vectors of labels, which empowers the network to map features into a semantic space. Specifically, $\mathbf{W}_{past}$ and $\mathbf{W}_{new}^k$ represent the matrices composed of past-label semantic embeddings and new-label semantic embeddings (learned from unsupervised large text corpora), respectively. In streaming label learning fashion (You et al., 2016; Wang et al., 2020b), as a preliminary, a feature extractor $\mathcal{F}$ with output matrix $\mathbf{W}_{past}$ is obtained by observing past-label data. A feature $\boldsymbol{z}$ produced by $\mathcal{F}$ would be close to the corresponding labels in the semantic space. To model the newly-arrived labels using only a few examples, we design a semantic inference mechanism to recognize and distinguish the new labels by leveraging the label-wise meta-knowledge. The semantic inference mechanism contains three different levels: 1) *feature-level*, utilizing proximity between features and labels' semantic representation; 2) *label-level*, exploiting semantic correlation between new labels and past labels; 3) *attention-level*, transferring the weighted probabilistic prediction from past labels to new labels. These three levels are convexly combined by learnable coefficients. Moreover, a novel label decision module $\phi(\cdot)$ is proposed with a meta-threshold loss to find the optimal thresholds value for each label by leveraging feature content.

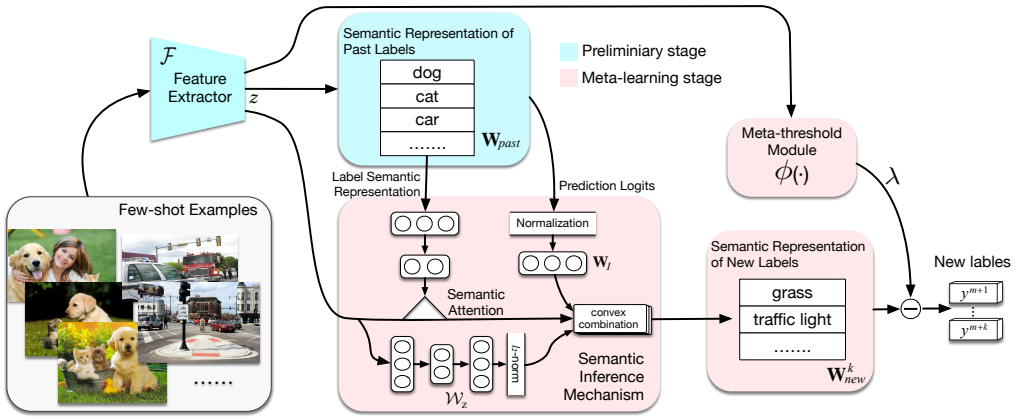


Figure 1: Semantic Inference Network (SIN) for Few-shot Streaming Label Learning. In the preliminary stage, we learn a semantic-aware feature extractor to build a semantic embedding space. In the meta-learning stage, we propose the semantic inference mechanism to exploit the correlation between past labels and new labels. A meta-threshold module generates optimal threshold values for each label. SIN is optimized by meta-learning to incorporate label-wise meta-knowledge.

3.2.1 Semantic Inference Mechanism

A key challenge of few-shot streaming label learning lies in the low-data problem causing overfitting and generalization issues. We propose a semantic inference mechanism to explore the correlation of features and labels in semantic space and acquire label-wise meta-knowledge, which could alleviate the model's overfitting and improve generalization. The semantic inference mechanism contains three different levels: feature-level, label-level, and attention-level, described in detail as follows.

Assume $\mathbf{W}_{past}=[\boldsymbol{w}^1_{past},...,\boldsymbol{w}^m_{past}]\in\mathbb{R}^{d\times m}$ is the matrix composed of m past-label semantic vectors, and $\mathbf{W}^k_{new}=[\boldsymbol{w}^1_{new},...,\boldsymbol{w}^k_{new}]\in\mathbb{R}^{d\times k}$ is the matrix composed of k new-label semantic vectors, where d is the dimension of semantic vectors. Note that the past label set L_{past} and the new label set L_{new} are disjoint. As the preliminary, we have a past-label classifier which consists of a feature extractor $\mathcal{F}$ and the output matrix $\mathbf{W}_{past}$ trained on $\mathcal{D}$. A feature $\boldsymbol{z}\in\mathbb{R}^{1\times d}$ is produced by $\mathcal{F}$, i.e., $\boldsymbol{z}=\mathcal{F}(\boldsymbol{x})$.

Feature-level. Since we have obtained features embedded into the semantic space, the output feature would naturally be closer to those with similar semantic meanings. In that case, we propose to infer new labels' prediction by feature-level proximity between $\boldsymbol{z}$ and new-label semantic vectors $\mathbf{W}^k_{new}$. Moreover, to learn the label-specific knowledge, we build a multi-layer perceptron $\mathcal{W}_Z$ to obtain transformation from $\boldsymbol{z}$ to new labels. The feature-level semantic inference I_f takes the form:

$$I_f(\boldsymbol{z})=\frac{\mathcal{W}_Z(\boldsymbol{z})}{\|\mathcal{W}_Z(\boldsymbol{z})\|_2}\mathbf{W}^k_{new}\ , \tag{1}$$

where $\mathcal{W}_Z(\boldsymbol{z})$ denotes a nonlinear transformation for $\boldsymbol{z}$, and $\|\cdot\|_2$ denotes l_2-norm which could eliminate the influence of the absolute magnitudes of semantic features and improve the robustness.

Label-level. Intuitively, we find that utilizing the semantic correlation between new labels and past labels as the meta-knowledge can help model to adapt new labels using only a few examples. For instance, a feature is considered to be associated with "car" in the past label space, and if we know the correlation between "car" and a new label "truck", then this can help us to classify the feature correctly in the new task. Generally, we use $\mathbf{W}^T_{past}\boldsymbol{w}^j_{new}$ to represent the correlation of the j-th new label in terms of m past labels. In order to learn the deep correlation between new labels and past labels, we define an explicit label-level semantic inference mechanism I_c as the following form:

$$I_c(\boldsymbol{z})=\frac{\boldsymbol{z}\mathbf{W}_{past}}{\|\boldsymbol{z}\mathbf{W}_{past}\|_2}\mathbf{W}_I\mathbf{W}^k_{new}\ , \tag{2}$$

where $\mathbf{W}_I\in\mathbb{R}^{m\times d}$ is a learnable matrix trained on different labels to learn the correlation between new and past labels. Although $\mathbf{W}_I$ is a linear operation, the l_2-norm can offer a non-linear operation. Since the number of past labels may be different between training and testing (see section 3.3), l_2-norm could also limit the adverse effects of absolute magnitudes fluctuations of $\boldsymbol{z}\mathbf{W}_{past}$.

Attention-level. Moreover, we design an attention-level semantic inference to transfer the probabilistic predictions on past labels to new labels. Assume that $\hat{y}^j$ is the probabilistic prediction for

j-th label and $\hat{y}^j \boldsymbol{w}^j_{past}$ is the corresponding probabilistic weighted semantic vector. By defining the convex combination of the past-label semantic embeddings: $e(\boldsymbol{x})=\sum_{j=1}^{m} \hat{y}^j \boldsymbol{w}^j_{past}=\mathbf{W}_{past}\hat{\boldsymbol{y}}=\mathbf{W}_{past}\text{sigmoid}(\mathcal{F}(\boldsymbol{x})W_{past})$, we could treat $e(\boldsymbol{x})$ as a region in the semantic space. Thus, the closer new label embedding $\boldsymbol{w}^j_{new}$ near the region $e(\boldsymbol{x})$, the higher probability of $\cos(e(\boldsymbol{x}),\boldsymbol{w}^j_{new})$ will be, where we use the cosine function to represent similarity. In that case, we can generate a probabilistic prediction of $\boldsymbol{x}$ on the corresponding new labels through this simple inference, i.e., $\cos(e(\boldsymbol{x}),\mathbf{W}^k_{new})=\cos(e(\boldsymbol{x}),[\boldsymbol{w}^1_{new},...,\boldsymbol{w}^k_{new}])$. To further improve the model's learning ability, we design the attention-level semantic inference mechanism based on the above basic idea. We consider an attentional head $a(\boldsymbol{z},\mathbf{W}_{past})$ with $\boldsymbol{z}$ to compute the query and past-label semantic vectors $\mathbf{W}_{past}$ used for keys and values. Next, we build a nonlinear network $\mathcal{W}_A$ to learn and figure out the relationship between the attention head $a(\boldsymbol{z},\mathbf{W}_{past})$ and new label semantic vectors $\mathbf{W}_{new}$. The attention-level semantic inference I_a can be formed as:

$$I_a(\boldsymbol{z})=\mathcal{W}_A\left(\frac{a(\boldsymbol{z},\mathbf{W}_{past})}{\|a(\boldsymbol{z},\mathbf{W}_{past})\|_2}\right)\mathbf{W}^k_{new}\,. \tag{3}$$

The implementation details of $a(\boldsymbol{z},\mathbf{W}_{past})$ can be found in Appendix A.2. Finally, we establish the semantic inference mechanism $\mathcal{I}$ as a convex combination of the above three levels in (1)-(3):

$$\mathcal{I}(\boldsymbol{z})=\mathcal{I}(\mathcal{F}(\boldsymbol{x}))=\gamma_f I_f(\boldsymbol{z})+\gamma_c I_c(\boldsymbol{z})+\gamma_a I_a(\boldsymbol{z}), \tag{4}$$

where γ_f, γ_c, and γ_a are the learnable module coefficients. The ablation study in Section 5.3 verifies the effectiveness of the semantic inference mechanism $\mathcal{I}$ and its three levels, respectively.

3.2.2 META-THRESHOLD

Different from traditional few-shot learning, which outputs only one label for an example, FSLL requires recognize and predict multiple newly-emerged labels. Hence, the *label decision* part is to determine which labels should appear in the prediction results from the list of predictive probability. Most existing methods apply simple heuristics for label decision, e.g., a threshold of 0.5 is used to all labels to generate the prediction decision. However, these methods ignore feature content when making the decision. We propose a novel meta-threshold module $\phi(\cdot)$ to find the optimal threshold values for each label by leveraging feature content. We define $\phi(\cdot)$ as a multi-layer perceptron which outputs a vector of label confidence threshold $\boldsymbol{\lambda}\in\mathbb{R}^k$ for making the decision: $\mathbb{1}_{(\mathcal{I}_j(\boldsymbol{z})>\lambda_j)}$, where $\mathbb{1}_{event}$ denotes the indicator function for $event$, $\forall j\in\{1,...,k\}$. To learn $\boldsymbol{\lambda}$ for each label, we propose a meta-threshold loss:

$$l_{thresh}(\boldsymbol{y}^{new},\mathcal{I}(\boldsymbol{z}),\boldsymbol{\lambda})=\sum_{j=1}^{k}\log\big(\cosh(\text{sigmoid}(\mathcal{I}_j(\boldsymbol{z})-\lambda_j)-y^{new}_j)\big), \tag{5}$$

where $cosh(x)=\frac{e^x+e^{-x}}{2}$. To independently learn the meta-thresholds of each label, we fix parameters of the semantic inference mechanism and optimize only for the parameters in $\phi(\cdot)$. Empirical studies indicate that meta-threshold with the sequential training improves the classification accuracy of FSLL.

3.3 TRAINING PROCEDURE

In order to learn the semantic inference network for few-shot streaming label learning, we use a training set $\mathcal{D}$ with m past labels as the sole input. First, we can obtain a past-label classifier (i.e., the feature extractor $\mathcal{F}$ as well as the output matrix $\mathbf{W}_{past}$) by minimizing a cross-entropy loss. Second, we employ a gradient-based meta-learning training (Finn et al., 2017; Antoniou et al., 2019) for the learnable parameters θ of SIN model (feature extractor $\mathcal{F}$ is frozen). To learn the label-wise meta knowledge, in each batch $\mathcal{T}_i$, we randomly extract k *simulative* new labels out from m past labels, and we treat them in the same way to simulate the newly-emerged labels in the testing procedure. Specifically, we sample N_s associated training examples per *simulative* new label (typically N_s<5) as the *support set* $\mathcal{S}$. The parameters are adapted to label-specific parameters θ' by applying a step of gradient descent on $\mathcal{S}$. Then, we sample other N_q associated examples per *simulative* new label as the *query set* $\mathcal{Q}$. The parameters θ are optimized by back-propagating in order to reduce errors produced by θ' on $\mathcal{Q}$. We iteratively train the model on different batches of *simulative* new labels. The meta objective is minimized to optimize the initial parameters θ, which contains the label-wise meta knowledge. More implementation details of the training procedure are provided in Appendix A.3.

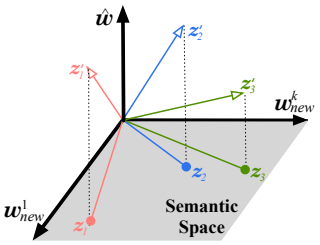


Figure 2: Features are projected to semantic space.

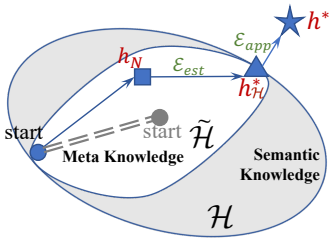


Figure 3: SIN solves the FSLL problem.

4 Generalization Analysis

In this section, we analyze the generalization of our learning model based on error decomposition in machine learning (Bottou & Bousquet, 2008; Bottou et al., 2018). Our analysis demonstrates that SIN leverages semantic knowledge and meta-knowledge to reduce the size of the hypotheses space and provide a good initialization for modeling new labels, bringing better generalizability of our proposed model (illustrated in Figure 2 and Figure 3).

For notational simplicity, let $P(\boldsymbol{x},\boldsymbol{y}^{new})$ be the ground-truth joint probability distribution of example $\boldsymbol{x}$ and label vector $\boldsymbol{y}^{new}$, and h^* be the optimal hypothesis from $\boldsymbol{x}$ to $\boldsymbol{y}^{new}$. Assume that the number N of the training examples for new labels is small. To discover h^*, the FSLL model determines a hypotheses space $\mathcal{H}$ containing a family of hypotheses h's by fitting the dataset. The performance of h is measured by a loss function $\ell(h(\boldsymbol{x}),\boldsymbol{y}^{new})$ defined over the prediction $\hat{\boldsymbol{y}}^{new}=h(\boldsymbol{x})$ and the real $\boldsymbol{y}^{new}$. Our benchmark is the optimal hypothesis h^* that minimizes the *expected risk*:

$$R(h)=\int \ell(h(\boldsymbol{x}),\boldsymbol{y}^{new})dP(\boldsymbol{x},\boldsymbol{y}^{new})=\mathbb{E}[\![\ell(h(\boldsymbol{x}),\boldsymbol{y}^{new})]\!], \tag{6}$$

that is, $h^*(\boldsymbol{x})=\text{argmin}_{\hat{\boldsymbol{y}}^{new}}\mathbb{E}[\![\ell(\hat{\boldsymbol{y}}^{new},\boldsymbol{y}^{new})|\boldsymbol{x}]\!]$. Since $P(\boldsymbol{x},\boldsymbol{y}^{new})$ is unknown, we define the *empirical risk* (by averaging the losses on the training set of N examples),

$$R_N(h)=\frac{1}{N}\sum_{i=1}^{N}\ell(h(\boldsymbol{x}_i),\boldsymbol{y}_i^{new}), \tag{7}$$

to approximate $R(h)$. The learning model aims to find the hypothesis $h_N=\text{argmin}_{h\in\mathcal{H}}R_N(h)$ that minimizes the empirical risk. As h^* is unlike to exist in the space $\mathcal{H}$, we define $h^*_{\mathcal{H}}=\text{argmin}_{h\in\mathcal{H}}R(h)$ to be the best approximation for h in $\mathcal{H}$. For simplicity, we assume that h^*, $h^*_{\mathcal{H}}$, and h_N are unique. The excessive error can be decomposed as (Bottou & Bousquet, 2008; Wang et al., 2020a):

$$\mathbb{E}[\![R(h_N)-R(h^*)]\!]=\mathbb{E}[\![R(h^*_{\mathcal{H}})-R(h^*)]\!]+\mathbb{E}[\![R(h_N)-R(h^*_{\mathcal{H}})]\!]=\mathcal{E}_{app}+\mathcal{E}_{est}, \tag{8}$$

where the expectation is regarding the random selection of the N training examples. As illustrated in Figure 3, the *approximation error* $\mathcal{E}_{app}$ measures how close the hypothesis in $\mathcal{H}$ can approximate the optimal hypothesis h^*, and the *estimation error* $\mathcal{E}_{est}$ measures the effect of minimizing the empirical risk $R_N(h)$ instead of the expected risk $R(h)$ within $\mathcal{H}$. In FSLL, since the number of annotated examples of new labels is very limited, the empirical risk $R_N(h)$ may be far away from being a good approximation of the expected risk $R(h)$, leading to the overfitting of empirical risk minimizer h_N.

To solve the above issue in FSLL, SIN leverages semantic knowledge and meta-knowledge (as described in Section 3), and we demonstrate that these can help reduce the estimation error $\mathcal{E}_{est}$ to improve the model's generalizability through the following two aspects. **(1)** SIN employs label semantic embeddings as the output layer weights $\mathbf{W}_{new}$, which can project features into a structured semantic space. As shown in Figure 2, a feature $\boldsymbol{z}'_i$ is mapped into the semantic space based on $\boldsymbol{w}^1_{new}$ and $\boldsymbol{w}^2_{new}$ as $\boldsymbol{z}_i$, and $\hat{\boldsymbol{w}}$ is the basis of the orthocomplement space. The semantic space essentially serves as the prior knowledge for the new labels. SIN uses the prior semantic knowledge to constrain $\mathcal{H}$ to a smaller hypothesis space $\tilde{\mathcal{H}}$ via the prior semantic knowledge. As shown in Figure 3, the gray area is excluded for optimization as it is considered to be unlikely to contain the best $h^*_{\mathcal{H}}$ according to semantic knowledge. For smaller $\tilde{H}$, the empirical risk $R_N(h)$ can be more approximate the expected risk $R(h)$, and the risk of overfitting is reduced. **(2)** Since we employ a meta-learning paradigm to learn from different labels, a good initialization with label-generic information is saved in the model as meta-knowledge. SIN leverages the meta-knowledge to obtain a good initial hypothesis h as a start to search the best $h^*_{\mathcal{H}}$ in $\tilde{\mathcal{H}}$. As shown in Figure 3, the gray *start* generated by meta-knowledge is closer to $h^*_{\mathcal{H}}$. Hence, $\mathcal{E}_{est}$ could be reduced by taking fewer optimization iterations with a few examples of new labels. Based on the above analysis, we give the following theorem.

Table 1: Few-shot streaming label classification accuracy (F1 score) on Delicious with 95% confidence intervals. k-way N_s-shot denotes k new labels with N_s tagged examples per label for training.

Method	5-way 1-shot	5-way 5-shot	10-way 1-shot	10-way 5-shot
Streaming label learning baselines				
SLL (You et al., 2016)	17.59±1.22%	29.27±0.67%	7.84±1.59%	17.53±0.86%
DSLL (Wang et al., 2020b)	24.42±1.84%	37.63±0.96%	11.22±1.47%	25.68±0.82%
Few-shot learning baselines				
MAML (Finn et al., 2017)	30.78±1.82%	40.76±0.95%	15.77±0.52%	27.69±0.37%
ProtoNet (Snell et al., 2017)	36.09±0.78%	42.53±0.64%	25.42±0.71%	31.26±0.59%
Reptile (Nichol et al., 2018)	29.09±0.13%	40.14±0.16%	19.53±0.21%	28.15±0.35%
ATAML (Jiang et al., 2018)	35.53±0.82%	43.51±0.75%	20.82±0.67%	30.59±0.59%
MAML++ (Antoniou et al., 2019)	41.21±0.64%	44.51±0.55%	23.91±0.57%	29.33±0.47%
LEO (Rusu et al., 2019)	42.52±0.16%	45.07±0.11%	30.95±0.14%	33.01±0.08%
Semantic-augment baselines				
DSLL (+semantic embeddings)	29.15±0.85%	40.26±0.25%	15.63±0.74%	26.94 ±0.51%
MAML (+semantic embeddings)	37.94±1.27%	44.03±0.62%	23.51±0.91%	32.56±0.84%
AM3 (Xing et al., 2019)	44.67±0.61%	47.44±0.47%	32.09±0.53%	34.73±0.43%
SIN (ours)	**46.17±0.35%**	**49.19±0.27%**	**35.06±0.31%**	**38.35±0.18%**

Theorem 1. *Assume the loss function ℓ is bounded and C-Lipschitz. Let h be a hypothesis in terms of k new labels using formulation $h_N=\operatorname{argmin}_{h\in\mathcal{H}} R_N(h)$ over a set of N training examples. Then with probability at least $1-\delta$, we have*

$$R(h_N)-\inf_{h\in\mathcal{H}} R(h)\leq C\mathfrak{R}_N(\mathcal{H})+\mathcal{O}\left(k\sqrt{\frac{\log(1/\delta)}{N}}\right),$$

where $\mathfrak{R}_N(\mathcal{H})=\mathbb{E}[\![\sup_{h\in\mathcal{H}}\frac{2}{N}\sum_i \epsilon_i h(\boldsymbol{x}_i)]\!]$ is the Rademacher complexity of $\mathcal{H}$, and ϵ_i is a random Rademacher variable: $prob(\epsilon_i=-1)=prob(\epsilon_i=1)=1/2$.

Refer to Appendix B for the proof. Theorem 1 relates the generalization error of SIN to the Rademacher complexity of the hypotheses space $\mathcal{H}$. The smaller the hypotheses space $\mathcal{H}$, the more generalizable the result is. Therefore, it turns out that SIN has the better generalizability by employing semantic knowledge and meta-knowledge to reduce the size of the hypotheses space.

5 Experiments

In this section, we conduct extensive experiments to evaluate the proposed model, SIN, in dealing with new labels under data scarcity. Experimental results show that SIN outperforms the state of the art of three different types of baselines: streaming label learning methods, few-shot learning methods, semantic-augment methods. Ablation studies verify the effectiveness of each module in SIN.

5.1 Experimental Setup

We use two widely used real-world multi-label datasets: Delicious (Tsoumakas et al., 2008) and Mir-Flickr (Huiskes & Lew, 2008) from text and image domains, respectively. Delicious has 12,920 training examples and 3,185 testing examples with 983 labels. Mir-Flickr has 20,000 training image features and 5,000 testing image features with 23 labels. We randomly choose 60% labels as past labels, 20% labels as new labels and the rest of labels as validation. Models are trained on the training set with past labels, and evaluated on the testing set with unseen new labels using only a few labeled examples. We use GloVe (Pennington et al., 2014) to generate the word vectors for the category labels as the semantic embeddings. The GloVe model is trained with large unsupervised text corpora. More details about the experimental setup can be found in Appendix C.

Baselines. We compare SIN with three families of methods. The first is streaming label learning methods: SLL (You et al., 2016) and DSLL (Wang et al., 2020b). The second fold is few-shot learning methods such as: MAML (Finn et al., 2017), ProtoNet (Snell et al., 2017), Reptile (Nichol et al., 2018), ATAML (Jiang et al., 2018), MAML++ (Antoniou et al., 2019), LEO (Rusu et al., 2019). Note

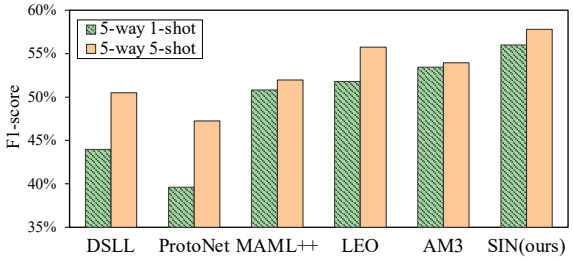


Figure 4: Performance comparison on Mir-Flickr

Student	5-way 1-shot	5-way 5-shot
SIN$\setminus l_2$	42.35±1.91%	44.47±1.26%
SIN$\setminus \phi$	45.25±0.83%	48.47±0.35%
SIN$\setminus \mathcal{I}$	39.12±1.17%	45.93±0.62%
SIN$\setminus I_f$	45.01±0.47%	47.93±0.34%
SIN$\setminus I_c$	45.23±0.28%	48.51±0.19%
SIN$\setminus I_a$	44.83±0.31%	47.64±0.21%
SIN	**46.17±0.35%**	**49.19±0.27%**

Table 2: Ablation study of components

that we improve and enable few-shot learning methods to handle the scenario of multiple new labels. To evaluate the effect of semantic embeddings more comprehensively and fairly, in the third fold baselines, we extend DSLL and MAML with the semantic representation of labels and compare with AM3 (Xing et al., 2019). AM3 also leverages semantic information of labels and achieves the the current state of the art among few-shot learning methods. For more fairness, all methods use the same feature extractor (same as SIN's) as the backbone of models. Details of baselines in Appendix C.3.

5.2 Results

The few-shot streaming label classification performance for SIN and other baselines are shown in Table 1 and Figure 4. We evaluate different numbers of new labels (-way) with different numbers of examples (-shot) on Delicious and MIR-Flickr measured by Micro F1-score and AUC (Wu & Zhou, 2017). First, the results show that SIN significantly outperforms other methods and has state-of-the-art performance on FSLL tasks. This indicates that the semantic knowledge and meta-knowledge leveraged by SIN boost the few-shot streaming label learning very effectively. Second, traditional streaming label learning and few-shot learning baselines (e.g., DSLL and MAML) get a significant performance improvement when embedded semantic information, and AM3 also leverages the textual features to improve classification accuracy. The results demonstrate the importance of semantic representation for FSLL. However, these semantic-augment baselines ignore the correlation between past labels and new labels. Third, it is worth noting that streaming label learning baselines perform worse than current few-shot learning baselines when examples are scarce, but they could recover quickly as examples increase. More detailed experimental results are given in Appendix D.

5.3 Ablation Study

To assess the effects of the proposed components in SIN, we perform the ablation study in Table 2. We first evaluate the impact of l_2-norm in the semantic inference. If SIN removes the l_2 normalization (denoted as SIN$\setminus l_2$), the performance of the model will be degraded. Since the l_2-norm could limit the adverse effects of absolute value fluctuations in semantic space, then the semantic similarity can be better represented. SIN$\setminus \phi$ denotes that the model uses a fixed global threshold for all labels, whose result shows the positive impact of the proposed meta-threshold module. The semantic inference mechanism is the key to our model. If we remove it from SIN (denoted as SIN$\setminus \mathcal{I}$), SIN would be reduced to a MAML algorithm with semantic embedding and produce a large performance degradation. The results confirm that the semantic inference mechanism can help boost the performance of FSLL. Specifically, we also separately assess the effects of the three different levels in the semantic inference mechanism (SIN$\setminus I_f$, SIN$\setminus I_c$, and SIN$\setminus I_a$ denote SIN without feature-level, correlation-level, and attention-level semantic inference, respectively). The results demonstrate effectiveness of each level.

6 Conclusion

This paper proposes a meta-learning framework, Semantic Inference Network (SIN), for the few-shot streaming label learning, which can effectively model new labels with only a few labeled examples. SIN exploits the semantic correlation between past labels and new labels and acquires label-wise meta-knowledge. Moreover, SIN incorporating a label decision module can find the optimal confidence threshold for each new label. Theoretical analysis proves that SIN can leverage semantic knowledge and meta-knowledge to reduce the size of the hypotheses space, resulting in better generalizability. Experiments show that SIN significantly outperforms state of the art on few-shot streaming label classification, and ablation studies indicate the effectiveness of the proposed components in SIN.

REFERENCES

Antreas Antoniou, Harrison Edwards, and Amos Storkey. How to train your MAML. In *International Conference on Learning Representations (ICLR)*, 2019.

Léon Bottou and Olivier Bousquet. The tradeoffs of large scale learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 161–168. 2008.

Léon Bottou, Frank E. Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.

Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. In *International Conference on Learning Representations (ICLR)*, 2019.

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

Thomas G. Dietterich. Steps toward robust artificial intelligence. *AI Magazine*, 38:3–24, 2017.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning (ICML)*, volume 70, pp. 1126–1135, 2017.

Chelsea Finn, Kelvin Xu, and Sergey Levine. Probabilistic model-agnostic meta-learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 9537–9548, 2018.

Sara Van De Geer. Applications of empirical process theory. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 51:416–417, 01 2002.

Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

Mark J. Huiskes and Michael S. Lew. The mir flickr retrieval evaluation. In *ACM International Conference on Multimedia Information Retrieval*, 2008.

Ray Jackendoff. On beyond zebra: The relation of linguistic and visual information. *Cognition*, 26 (2):89 – 114, 1987.

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.

Georg Krempl, Indre Žliobaite, Dariusz Brzeziński, Eyke Hüllermeier, Mark Last, Vincent Lemaire, Tino Noack, Ammar Shaker, Sonja Sievi, Myra Spiliopoulou, and Jerzy Stefanowski. Open challenges for data stream mining research. *SIGKDD Explor. Newsl.*, 16(1):1–10, 2014.

Michel Ledoux and Michel Talagrand. *Probability in Banach Spaces: Rademacher Averages*, pp. 89–121. Springer Berlin Heidelberg, 1991.

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.

Andreas Maurer. A vector-contraction inequality for rademacher complexities. In *Algorithmic Learning Theory*, pp. 3–17, 2016.

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

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)*, pp. 1532–1543, 2014.

David Pollard. Empirical processes: Theory and applications. *NSF-CBMS Regional Conference Series in Probability and Statistics*, 2:i–86, 1990.

Vijay Raghavan and Alaaeldin Hafez. Dynamic data mining. In *Intelligent Problem Solving. Methodologies and Approaches*, pp. 220–229. Springer, 2000.

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 (ICLR)*, 2019.

Claude Sammut and Geoffrey I. Webb (eds.). *Encyclopedia of Machine Learning: McDiarmid's Inequality*, pp. 651–652. 2010.

Linda Smith and Michael Gasser. The development of embodied cognition: Six lessons from babies. *Artificial Life*, 11(1–2):13–30, 2005.

Linda B. Smith. Learning to recognize objects. *Psychological Science*, 14(3):244–250, 2003. PMID: 12741748.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 4077–4087. 2017.

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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

S. Thrun. Lifelong learning algorithms. In *Learning to Learn*, pp. 181–209, 1998.

Grigorios Tsoumakas, Ioannis Katakis, and Ioannis P. Vlahavas. Effective and efficient multilabel classification in domains with large number of labels. In *ECML/PKDD 2008 Workshop on Mining Multidimensional Data (MMD)*, 2008.

V. N. Vapnik. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10 (5):988–999, 1999.

Vladimir N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1998.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, koray kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 3630–3638. 2016.

Yaqing Wang, Quanming Yao, James T. Kwok, and Lionel M. Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv.*, 53(3), June 2020a.

Zhen Wang, Liu Liu, and Dacheng Tao. Deep streaming label learning. In *International Conference on Machine Learning (ICML)*, pp. 378–387. 2020b.

Xi-Zhu Wu and Zhi-Hua Zhou. A unified view of multi-label performance measures. In *International Conference on Machine Learning (ICML)*, volume 70, pp. 3780–3788, 2017.

Chen Xing, Negar Rostamzadeh, Boris Oreshkin, and Pedro O O. Pinheiro. Adaptive cross-modal few-shot learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 4847–4857. 2019.

Shan You, Chang Xu, Yunhe Wang, Chao Xu, and Dacheng Tao. Streaming Label Learning for Modeling Labels on the Fly. *arXiv preprint arXiv:1604.05449*, 2016.

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

Pengchuan Zhang, Qiang Liu, Dengyong Zhou, Tao Xu, and Xiaodong He. On the discrimination-generalization tradeoff in GANs. In *International Conference on Learning Representations (ICLR)*, 2018.

Zhi-Hua Zhou. Learnware: On the future of machine learning. *Front. Comput. Sci.*, 10(4):589–590, 2016.

APPENDIXES

Appendix A supplements the content of motivation, attention-level semantic inference and training procedure. Appendix B details the proof of Theorem 1. Appendix C presents the implementation details and definition of evaluation metrics. Appendix D presents additional empirical evaluation for few-shot streaming label learning and further analyses.

A METHOD

A.1 MOTIVATION

We illustrate an instance of FSLL, as shown in Figure 5, when a photo is posted to Facebook or Twitter, the photo may be continuously tagged with new labels by users who are browsing, and the classification system needs to be updated accurately according to the new labels (Dietterich, 2017).

We hypothesize that in the context of few-shot streaming label learning, semantic representation from the text can be a powerful source of information to distinguish new labels. As illustrated in Figure 6, the semantic representation naturally brings correlation across labels, e.g., the label "dog" is semantically close to the label "Labrador" in the word embedding space. The semantic correlation across labels would help to structure and regularize the overwhelming output space of FSLL.

Figure 5: A photo is being tagged with new labels

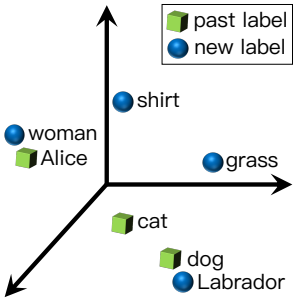


Figure 6: Labels in semantic space

A.2 ATTENTION-LEVEL SEMANTIC INFERENCE

We design an attention-level semantic inference to transfer the probabilistic predictions on past labels to new labels. A basic idea of this mechanism is using the output score of past labels to compute a weighted combination of new labels embeddings in the semantic space. Given example $\boldsymbol{x}$ and past-label classifier, we can get a label prediction vector $\hat{\boldsymbol{y}}=[\hat{y}^1,...,\hat{y}^m]$, where $\hat{y}^j$ is the probabilistic prediction for the j-th label. Consequently, $\hat{y}^j\boldsymbol{w}^j_{past}$ is the probabilistic weighted word vector for the j-th past label. More formally, by defining the convex combination of the past-label word embeddings:

$$e(\boldsymbol{x})=\sum_{j=1}^{m}\hat{y}^j\boldsymbol{w}^j_{past}=\mathbf{W}_{past}\hat{\boldsymbol{y}}=W_{past}\text{sigmoid}(\mathcal{F}(\boldsymbol{x})W_{past}),$$

In that case, we can generate a probabilistic prediction of $\boldsymbol{x}$ on the corresponding new labels through this simple inference, i.e., $\cos(e(\boldsymbol{x}),\mathbf{W}^k_{new})=\cos(e(\boldsymbol{x}),[\boldsymbol{w}^1_{new},...,\boldsymbol{w}^k_{new}])$. However, the simple inference mentioned above has not taken the full utilization of label-specific knowledge. In order to improve the learning ability of the model, we further design the attention-level semantic inference mechanism based on the above basic idea. We consider an attentional head $a(\boldsymbol{z},\mathbf{W}_{past})$ with $\boldsymbol{z}$ to compute the query and base-label vectors used for keys and values, which takes the form:

$$a(\boldsymbol{z},\mathbf{W}_{past})=\frac{\text{sigmoid}\left(\frac{q(\boldsymbol{z})\mathbf{W}_{past}}{\tau}\right)}{\left\|\text{sigmoid}\left(\frac{q(\boldsymbol{z})\mathbf{W}_{past}}{\tau}\right)\right\|_1}\mathbf{W}^T_{past}\,, \tag{9}$$

Algorithm 1 Semantic Inference Network: Meta-learning Training

Require: Training set D
Require: Learning rates α, β
Train feature extractor $\mathcal{F}$ on $\mathcal{D}$
Initialize θ ▷ Initialize all parameters
while not done **do**
  Sample batch of new labels tasks $\mathcal{T}_i \sim \mathcal{D}$ ▷ 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**
    $\theta_i' = \theta - \alpha \nabla_\theta \mathcal{L}(\mathcal{S}_{\mathcal{T}_i}; \theta)$ ▷ Compute temporary parameters
  end for
  Update $\theta \leftarrow \theta - \beta \nabla_\theta \sum_{\mathcal{T}_i} \mathcal{L}(\mathcal{Q}_{\mathcal{T}_i}; \theta_i')$ ▷ Update network parameters
end while

where $q(\boldsymbol{z}) = \boldsymbol{z}\mathbf{W}_q, \mathbf{W}_q \in \mathbb{R}^{d \times d}$ is a learnable matrix, $\|\cdot\|_1$ is l_1-norm, and τ is a temperature. Different from the traditional attention mechanism, we use the sigmoid function to compute the probability, rather than softmax function, causing that the sum of the probabilities is not 1. Hence, we perform l_1-norm to regularize different new-label tasks.

A.3 Training Procedure

In order to learn the semantic inference network for few-shot streaming label learning, we use as the sole input a training set $\mathcal{D}$ of m past labels.

First, we can obtain a past-label classifier (i.e., the feature extractor $\mathcal{F}$ as well as the output matrix W_{past}) by minimizing a cross-entropy loss of the following form:

$$l_{CE}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\sum_{j=1}^{m} \left[y^j \log(\hat{y}^j) + (1 - y^j) \log(1 - \hat{y}^j) \right].$$

Second, we employ a gradient-based meta-learning training (Finn et al., 2017; Antoniou et al., 2019) for the learnable parameters θ of the semantic inference mechanism and the meta-threshold model (the feature extractor is frozen during this stage). The meta-learning training is described in Algorithm 1. To learn the label-wise meta-knowledge, in each batch $\mathcal{T}_i$, we randomly extract k *simulative* new labels out from m past labels, and we treat them in the same way to simulate the unseen new labels in the testing procedure. Specifically, we sample N_s associated training examples per *simulative* new label (typically N_s<5) as the *support set* $\mathcal{S}$. The parameters are adapted to label-specific parameters θ' by applying a step of gradient descent on $\mathcal{S}$:

$$\theta_i' = \theta - \alpha \nabla_\theta \mathcal{L}(\mathcal{S}; \theta), \tag{10}$$

where α is a learning rate, $\mathcal{L}$ generates the loss. Then, we sample other N_q associated examples per *simulative* new label as the *query set* $\mathcal{Q}$. The parameters θ are optimized by back-propagating in order to reduce errors on the query set $\mathcal{Q}$:

$$\theta \leftarrow \theta - \beta \nabla_\theta \sum_{\mathcal{T}_i \sim \mathcal{D}} \mathcal{L}(\mathcal{Q}; \theta_i'), \tag{11}$$

where β is a learning rate. We iteratively train the model on different batches of *simulative* new labels. The meta objective (11) is minimized to optimize the initial parameters θ, which contains the label-wise meta-knowledge.

B Proof of Theorem 1

In the section, we shall provide proof of Theorem 1. Our proof demonstrates that the smaller the hypotheses space $\mathcal{H}$, the more generalizable the result is. It turns out that the proposed SIN has better generalizability by employing semantic knowledge and meta-knowledge to reduce the size

of the hypotheses space. In few-shot streaming label learning (FSLL), newly emerged labels are associated with only a few examples. Given N examples with k new labels, SIN learns a hypothesis h by minimizing the *empirical risk*,

$$R_N(h)=\frac{1}{N}\sum_{i=1}^{N}\ell(h(\boldsymbol{x}_i),\boldsymbol{y}_i^{new})=\mathbb{E}_N[\![\ell(h(\boldsymbol{x}),\boldsymbol{y}^{new})]\!], \tag{12}$$

where $\ell(\cdot)$ denotes the loss function; $\hat{\boldsymbol{y}}^{new}=h(\boldsymbol{x})$ is the prediction for $\boldsymbol{y}$. The learning model aims to find the hypothesis

$$h_N=\operatorname*{argmin}_{h\in\mathcal{H}} R_N(h)$$

that minimizes the empirical risk (12). The goal of the proof is to show that the excess risk $R(h_N)-R_N(h_N)$ is related to the size of the hypotheses space $\mathcal{H}$, where $R(h)$ is the *expected risk*, defined as:

$$R(h)=\int\ell(h(\boldsymbol{x}),\boldsymbol{y}^{new})dP(\boldsymbol{x},\boldsymbol{y}^{new})=\mathbb{E}[\![\ell(h(\boldsymbol{x}),\boldsymbol{y}^{new})]\!]=\mathop{\mathbb{E}}_{(\widetilde{\boldsymbol{x}}_i,\widetilde{\boldsymbol{y}}_i^{new})}\left[\!\!\left[\frac{1}{N}\sum_{i=1}^{N}\ell(h(\widetilde{\boldsymbol{x}}),\widetilde{\boldsymbol{y}}_i^{new})\right]\!\!\right]. \tag{13}$$

This can be achieved by the uniform concentration bounds developed in empirical process (Pollard, 1990; Geer, 2002) and statistical learning theory (Vapnik, 1998; 1999; Zhang et al., 2018). In particular, the size of hypotheses space $\mathcal{H}$ can be characterized by the Rademacher complexity of $\mathcal{H}$, defined as

$$\mathfrak{R}(\mathcal{H})=\mathop{\mathbb{E}}_{(\boldsymbol{x}_i,\epsilon_i)}\left[\!\!\left[\sup_{h\in\mathcal{H}}\frac{2}{N}\sum_{i}^{N}\epsilon_i h(\boldsymbol{x}_i)\right]\!\!\right], \tag{14}$$

where ϵ_i is a random Rademacher variable: $prob(\epsilon_i=-1)=prob(\epsilon_i=1)=1/2$.

By using McDiarmid's inequality (Sammut & Webb, 2010), the excess risk $R(h_N)-R_N(h_N)$ can be bounded by the expected supremum deviation of empirical risks,

$$\begin{aligned}
R(h_N)-R_N(h_N)&\le\sup_{h\in\mathcal{H}}\{R(h)-R_N(h)\}\\
&=\sup_{h\in\mathcal{H}}\{\mathbb{E}[\![\ell(h(\boldsymbol{x}),\boldsymbol{y}^{new})]\!]-\mathbb{E}_N[\![\ell(h(\boldsymbol{x}),\boldsymbol{y}^{new})]\!]\}\\
&=\sup_{h\in\mathcal{H}}\left\{\mathop{\mathbb{E}}_{(\widetilde{\boldsymbol{x}}_i,\widetilde{\boldsymbol{y}}_i^{new})}\left[\!\!\left[\frac{1}{N}\sum_{i=1}^{N}\ell(h(\widetilde{\boldsymbol{x}}),\widetilde{\boldsymbol{y}}_i^{new})\right]\!\!\right]-\frac{1}{N}\sum_{i=1}^{N}\ell(h(\boldsymbol{x}_i),\boldsymbol{y}_i^{new})\right\}\\
&\triangleq g((\boldsymbol{x}_1,\boldsymbol{y}_1^{new}),...,(\boldsymbol{x}_N,\boldsymbol{y}_N^{new})).
\end{aligned}$$

Since the decomposable loss function $\ell(h(\boldsymbol{x}_i),\boldsymbol{y}_i^{new})=\sum_{j=m+1}^{m+k}\ell(h^j(\boldsymbol{x}),y^j)$ are bounded, changing any $(\boldsymbol{x}_i,\boldsymbol{y}_i^{new})$ would induce a perturbation of $g((\boldsymbol{x}_1,\boldsymbol{y}_1^{new}),...,(\boldsymbol{x}_N,\boldsymbol{y}_N^{new}))$ at most $\mathcal{O}(\frac{k}{N})$. Then by applying McDiarmid's inequality, the sum of squared perturbations is bounded by $\frac{2k^2}{N}$, and thus the excess risk is bounded by a term related to the expectation of $g((\boldsymbol{x}_1,\boldsymbol{y}_1^{new}),...,(\boldsymbol{x}_N,\boldsymbol{y}_N^{new}))$, i.e., the expected supremum deviation. Therefore, we have established that with probability at least $1-\delta$,

$$R(h_N)-R_N(h_N)\le\mathop{\mathbb{E}}_{(\boldsymbol{x}_i,\boldsymbol{y}_i^{new})}[\![g((\boldsymbol{x}_1,\boldsymbol{y}_1^{new}),...,(\boldsymbol{x}_N,\boldsymbol{y}_N^{new}))]\!]+\mathcal{O}\left(k\sqrt{\frac{\log(1/\delta)}{N}}\right).$$

Next, we bound the expected supremum deviation by Rademacher complexity (Ledoux & Talagrand, 1991; Maurer, 2016). We have

$$
\begin{aligned}
&\mathop{\mathbb{E}}_{(\boldsymbol{x}_i,\boldsymbol{y}_i^{new})}[\![g((\boldsymbol{x}_1,\boldsymbol{y}_1^{new}),...,(\boldsymbol{x}_N,\boldsymbol{y}_N^{new}))]\!] \\
=&\mathop{\mathbb{E}}_{(\boldsymbol{x}_i,\boldsymbol{y}_i^{new})}\left[\!\!\left[\sup_{h\in\mathcal{H}}\{\mathbb{E}[\![\ell(h(\boldsymbol{x}),\boldsymbol{y}^{new})]\!]-\mathbb{E}_N[\![\ell(h(\boldsymbol{x}),\boldsymbol{y}^{new})]\!]\}\right]\!\!\right] \\
=&\mathop{\mathbb{E}}_{(\boldsymbol{x}_i,\boldsymbol{y}_i^{new})}\left[\!\!\left[\sup_{h\in\mathcal{H}}\left\{\mathop{\mathbb{E}}_{(\widetilde{\boldsymbol{x}}_i,\widetilde{\boldsymbol{y}}_i^{new})}\left[\!\!\left[\frac{1}{N}\sum_{i=1}^{N}\ell(h(\widetilde{\boldsymbol{x}}),\widetilde{\boldsymbol{y}}_i^{new})\right]\!\!\right]-\frac{1}{N}\sum_{i=1}^{N}\ell(h(\boldsymbol{x}_i),\boldsymbol{y}_i^{new})\right\}\right]\!\!\right] \\
\leq&\mathop{\mathbb{E}}_{\substack{(\boldsymbol{x}_i,\boldsymbol{y}_i^{new})\\(\widetilde{\boldsymbol{x}}_i,\widetilde{\boldsymbol{y}}_i^{new})}}\left[\!\!\left[\sup_{h\in\mathcal{H}}\left\{\frac{1}{N}\sum_{i=1}^{N}\ell(h(\widetilde{\boldsymbol{x}}),\widetilde{\boldsymbol{y}}_i^{new})-\frac{1}{N}\sum_{i=1}^{N}\ell(h(\boldsymbol{x}_i),\boldsymbol{y}_i^{new})\right\}\right]\!\!\right] \\
=&\mathop{\mathbb{E}}_{\substack{(\boldsymbol{x}_i,\boldsymbol{y}_i^{new})\\(\widetilde{\boldsymbol{x}}_i,\widetilde{\boldsymbol{y}}_i^{new}),\epsilon_i}}\left[\!\!\left[\sup_{h\in\mathcal{H}}\left\{\frac{1}{N}\sum_{i=1}^{N}\epsilon_i(\ell(h(\widetilde{\boldsymbol{x}}),\widetilde{\boldsymbol{y}}_i^{new})-\ell(h(\boldsymbol{x}_i),\boldsymbol{y}_i^{new}))\right\}\right]\!\!\right] \\
\leq&\mathop{\mathbb{E}}_{(\boldsymbol{x}_i,\boldsymbol{y}_i^{new}),(\widetilde{\boldsymbol{x}}_i,\widetilde{\boldsymbol{y}}_i^{new}),\epsilon_i}\left[\!\!\left[\sup_{h\in\mathcal{H}}\left\{\frac{1}{N}\sum_{i=1}^{N}\epsilon_i\ell(h(\widetilde{\boldsymbol{x}}),\widetilde{\boldsymbol{y}}_i^{new})\right\}\right]\!\!\right] \\
&+\mathop{\mathbb{E}}_{(\boldsymbol{x}_i,\boldsymbol{y}_i^{new}),(\widetilde{\boldsymbol{x}}_i,\widetilde{\boldsymbol{y}}_i^{new}),\epsilon_i}\left[\!\!\left[\sup_{h\in\mathcal{H}}\left\{\frac{1}{N}\sum_{i=1}^{N}\epsilon_i\ell(h(\boldsymbol{x}_i),\boldsymbol{y}_i^{new})\right\}\right]\!\!\right] \\
\leq&2\mathop{\mathbb{E}}_{(\boldsymbol{x}_i,\boldsymbol{y}_i^{new}),\epsilon_i}\left[\!\!\left[\sup_{h\in\mathcal{H}}\left\{\frac{1}{N}\sum_{i=1}^{N}\epsilon_i\ell(h(\boldsymbol{x}_i),\boldsymbol{y}_i^{new})\right\}\right]\!\!\right] \\
\leq&C\mathop{\mathbb{E}}_{(\boldsymbol{x}_i,\epsilon_i)}\left[\!\!\left[\sup_{h\in\mathcal{H}}\frac{2}{N}\sum_{i}^{N}\epsilon_i h(\boldsymbol{x}_i)\right]\!\!\right]:=C\mathfrak{R}(\mathcal{H}),
\end{aligned}
$$

which establishes Theorem 1. Note that the first inequality employs Jensen inequality. The second inequality is based on the convexity of the supremum function. The third inequality is based on (13). The last inequality under the assumption that the loss ℓ is bounded and C-Lipschitz.

C Experimental Settings

C.1 Implementation Details

Pytorch[2] is used to implement the proposed algorithm and 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 the training stage, we use Adam optimizer with a fixed learning rate of 0.001, weight decay 10^{-6}. We train models 100 epochs, where each epoch contains 200 tasks randomly sampled from training set $\mathcal{D}_{train}$. For the testing stage, we test models on 200 novel tasks randomly sampled from $\mathcal{D}_{test}$ to get average results. The semantic embedding model, GloVe (Pennington et al., 2014), 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 temperature hyperparameter in Equation (9) is set in the range [0.1, 10], and dropout rate in the networks is set in the range [0.1, 0.5]. We also provide the source code for reference[4].

[2] https://pytorch.org/

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

[4] https://github.com/ICLR-FSLL/SIN

C.2 EVALUATION METRICS AND SETTINGS

Given a query set $\mathcal{Q}_\mathcal{T}$ sampled from test dataset $\mathcal{D}_{test}$ 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 (example) and $\boldsymbol{y}_i\in\mathcal{Y}\subseteq\{0,1\}^{k\times 1}$ is the corresponding output new label vector ($i\in\{1,...,N\}$). 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 $Y_{i\cdot}^+$ ($Y_{i\cdot}^-$) 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 examples of the j-th label and $Y_{\cdot j}^-=\{i|y_i^j=0\}$ denotes the set of non-associated examples similarly. We use $|\cdot|$ to represent the cardinality of a set. For an k-way N_s-shot setting, each task is sampled with k labels, and each label includes N_s support examples and 13 query examples. Note that due to the label distribution of multi-label data (Zhang & Zhou, 2014), several labels may correspond to more examples than other labels, however, the comparison on different methods are fair because of the same setting.

Table 3: Definitions of streaming label learning performance measures.

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	$\textit{micro-AUC}(F)=\frac{\lvert\mathcal{S}_{\text{micro}}\rvert}{(\sum_{i=1}^{n}\lvert Y_{i\cdot}^+\rvert)\cdot(\sum_{i=1}^{n}\lvert Y_{i\cdot}^-\rvert)}$ $\mathcal{S}_{\text{micro}}=\{(a,b,i,j)\lvert(a,b)\in Y_{\cdot i}^+\times Y_{\cdot j}^-,\ f_i(\boldsymbol{x}_a)\geq f_j(\boldsymbol{x}_b)\}$	AUC averaging on prediction matrix. $\mathcal{S}_{\text{micro}}$ is the set of correct quadruples.

Table 3 summarizes two popular multi-label evaluation metrics used in this paper, which can be divided into a bipartition-based metric, i.e., micro-F1, and a ranking-based metric, i.e., AUC (Wu & Zhou, 2017). We assume that $H:\mathbb{R}^d\to\{0,1\}^m$ is the FSLL classifier and predicts which labels an example is associated. 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 FSLL 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

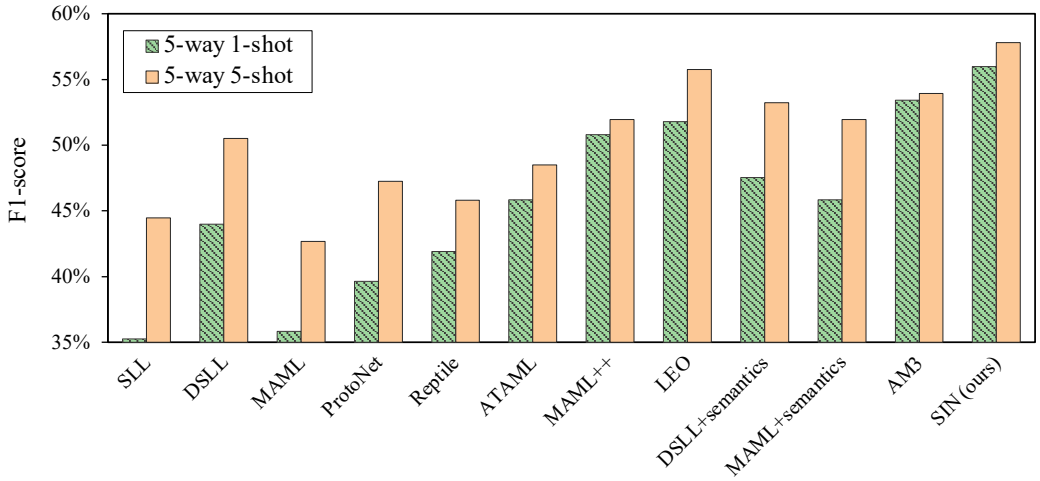
SLL (You et al., 2016) is a streaming label learning method that learns a mapping from past label vectors to new label vectors and assumes the new classifier can inherit the mapping relationship as a regularization. Constrained by the hypothesis, new classifiers can improve performance. DSLL (Wang et al., 2020b) is a DNN-based framework to learn deep relationships across labels with label smoothing techniques. Moreover, DSLL has the ability to distill the feature-level knowledge from a past-label classifier to a new-label classifier and outperforms other methods on streaming label learning. MAML (Finn et al., 2017) is a groundbreaking model-agnostic meta-learning framework for few-shot learning, which learns a good initialization for new tasks. Reptile (Nichol et al., 2018) proposes a *shortest descent* method to further improve efficiency and performance. ATAML (Jiang et al., 2018) introduces attention mechanism into meta-learning to learn task-agnostic representation. MAML++ (Antoniou et al., 2019) is a state-of-the-art meta-learning framework for the few-shot learning problem, which employs multi-step loss optimization to improve generalization performance. LEO (Rusu et al., 2019) applies pre-trained representations on a low-dimensional latent space instead of the original high-dimensional parameter space, which achieves better classification performance. Prototypical Network (Snell et al., 2017) can first train an average value of the features belonging to the same label in the metric space as the prototype and then perform the nearest neighbor classification.

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

Table 4: Few-shot streaming label classification accuracy (AUC) on Delicious with 95% confidence intervals. k-way N_s-shot denotes k new labels with N_s tagged examples per label for training.

Method	5-way 1-shot	5-way 5-shot	10-way 1-shot	10-way 5-shot
Streaming label learning baselines				
SLL (You et al., 2016)	60.11±1.81%	67.83±1.37%	61.10±1.53%	70.59±1.16%
DSLL (Wang et al., 2020b)	63.59±2.36%	70.91±1.31%	62.70±2.07%	72.87±1.38%
Few-shot learning baselines				
MAML (Finn et al., 2017)	68.24±2.02%	73.82±1.35%	68.17±1.01%	74.42±0.74%
ProtoNet (Snell et al., 2017)	65.96±1.53%	71.87±1.23%	69.79±1.45%	73.71±1.21%
Reptile (Nichol et al., 2018)	66.65±0.53%	70.39±0.61%	68.54±0.49%	73.79±0.55%
ATAML (Jiang et al., 2018)	70.14±1.54%	72.50±1.42%	68.68±1.20%	74.60±1.04%
MAML++ (Antoniou et al., 2019)	72.50±1.31%	74.09±1.13%	71.38±1.26%	73.85±0.92%
LEO (Rusu et al., 2019)	72.39±0.34%	74.87±0.26%	69.43±0.27%	74.10±0.18%
Semantic-augment baselines				
DSLL (+semantic embeddings)	67.35±1.75%	73.13±0.63%	68.38±1.39%	72.99 ±1.12%
MAML (+semantic embeddings)	71.43±1.91%	74.30±1.28%	70.03±1.54%	74.59±1.32%
AM3 (Xing et al., 2019)	72.58±1.31%	74.95±0.68%	72.25±1.13%	74.43±0.74%
SIN (ours)	**75.38±0.61%**	**76.41±0.51%**	**73.90±0.71%**	**76.13±0.47%**

Figure 7: Performance comparison on Mir-Flickr. k-way N_s-shot denotes k new labels with N_s tagged examples per label for training.



AM3 (Xing et al., 2019) introduces semantic representation to adapt the prototype and achieves state-of-the-art performance on few-shot learning.

D Additional Results and Analyses

D.1 Additional Results

Table 4 and Figure 7 show the average performance of the few-shot streaming label classification for SIN and baselines on Delicious and Mir-Flickr, respectively. The results consistently show that our model outperforms the baselines on AUC and F1-score metrics for both 1-shot and 5-shot, 5-way and 10-way. Please note that Mir-Flickr only has 5 new labels in the testing dataset. SIN 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 few-shot streaming label learning.

D.2 ADDITIONAL ANALYSES

Influence of semantic correlation. As shown in Section 5 and Appendix D.1, it can be confirmed that label semantic correlation is a key for few-shot streaming label learning. We leverage label semantic correlation in both of semantic-aware feature extractor $\mathcal{F}$ and semantic inference mechanism $\mathcal{I}$. Exploiting label correlation can facilitate FSLL 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 newly 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 few-shot streaming label learning. Table 2 shows that if we remove the semantic inference $\mathcal{I}$ (denoted as SIN$\backslash\mathcal{I}$), SIN will produce a significant performance degradation.

Influence of l_2-norm. Table 2 demonstrates that if remove the l_2 normalization (denoted as SIN$\backslash l_2$-norm), the performance of the model will be degraded. 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 feature-level inference (Equation (1)), l_2-norm is used to eliminate the influence of the absolute magnitudes of semantic features and improve the robustness. In label-level inference (Equation (2)), 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 past labels is different between training and testing (described in Section 3.3), by using l_2-norm, the absolute value of the feature transformation in training and testing can be kept consistent, which is important for the convergence of the model. In attention-level inference (Equation (3)), l_2-norm provides the regularity of the attention value.